

SLAIR: Structured Latent Flow Matching for All-in-One Image Restoration

Shuyi Liang^{*,1}, Yixin Yang^{*,1}, Hanyue Lou¹,
Yuning Cui², and Boxin Shi^{†1}

¹ Peking University

² Technical University of Munich

liangshuyi@stu.pku.edu.cn yuning.cui@in.tum.de
{yangyixin93,hylz,shiboxin}@pku.edu.cn

Abstract. Image restoration aims to recover high-quality images from degraded observations, yet existing methods struggle with diverse degradation types. In this work, we propose SLAIR, an all-in-one image restoration framework integrating latent space separation and deterministic transport. We introduce a latent-space separation approach that ensures stable, distinct representations of degraded and clean images, facilitating effective flow matching. By learning continuous transport flow in a latent space via an ordinary differential equation (ODE), our method enables deterministic restoration without relying on degradation labels or stochastic sampling. While the restoration is performed in the latent space, a high-frequency refinement module is introduced into the latent decoder to handle detail distortion during decoding. Experiments demonstrate that SLAIR outperforms existing approaches in terms of both perceptual quality and computational efficiency, making it a robust solution for scalable, degradation-agnostic image restoration. Our project page is available at <https://fightforql.github.io/SLAIR-page>.

Keywords: All-in-one Image Restoration · Flow Matching

1 Introduction

Image restoration aims to recover high-quality images from degraded observations and plays a crucial role in applications such as robotic vision, remote sensing, and medical imaging [14, 50, 59]. While task-specific models for denoising [67, 68], deraining [11, 42], and dehazing [21, 44] have achieved significant progress, practical deployment often requires a unified system capable of handling diverse, unknown, and even mixed degradations. This motivates the development of *all-in-one* image restoration models [8, 18, 22, 41, 46, 51, 64] that generalize across degradation types without prior knowledge.

* Equal contribution. † Corresponding author.

¹ State Key Laboratory of Multimedia Information Processing and National Engineering Research center of Visual Technology, School of Computer Science, Peking University

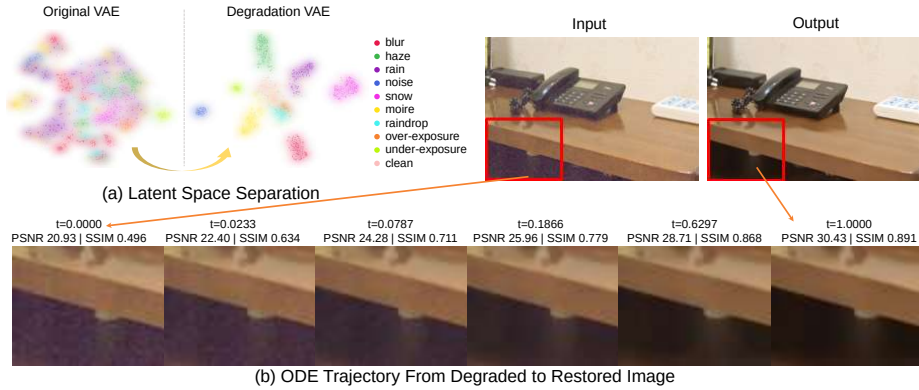


Fig. 1: Latent space visualization and performance analysis. (a) A comparison between the original Variational Autoencoder (VAE) [60] and our degradation-aware VAE using t-SNE [34]. Each point represents a single image encoded by the VAE and projected to 2-D via t-SNE, with colors denoting eight degradation types and clean. While the original space is entangled, our Degradation VAE effectively clusters distinct degradation types, providing a more structured prior for the restoration process. (b) Illustrates the sampling process via the flow ODE. As the time step t evolves from 0 to 1, the degraded input is progressively refined into a high-quality image.

Despite recent advances, existing all-in-one approaches still face several limitations. First, many methods explicitly estimate degradations [12, 24, 31] or rely on prompt-based modulation [41, 55], creating a dependency between *degradation perception* and *image restoration*. Once the perceived degradation is inaccurate, restoration is inevitably misled, leading to artifacts. Second, most approaches are trained with deterministic regression objectives [8, 41], which tend to average plausible solutions and suppress details. Third, diffusion-based models [5, 19] improve perceptual realism but rely on stochastic sampling and iterative refinement, increasing computational cost. Therefore, designing a unified framework that simultaneously achieves high perceptual quality, strong degradation adaptivity, and computational efficiency remains a challenging problem.

In this work, we revisit all-in-one image restoration from the perspective of *flow matching* [26]. Rather than predicting degradation features or reversing a stochastic corruption process, we formulate restoration as learning a conditional neural vector field that continuously transports degraded latents toward the clean latents. This perspective eliminates explicit degradation modeling and reframes restoration as distribution-level alignment instead of pixel-wise regression. A key requirement for stable transport is a well-structured relationship between degraded and clean representations. When their latent embeddings are entangled using the original Variational Autoencoder (VAE) encoder, as shown in the left side of Figure 1 (a), the transport dynamics become complex and difficult to learn. To address this issue, we introduce a VAE-based latent structuring mechanism that explicitly enlarges the representation gap between degraded

and clean samples. By increasing inter-manifold separation in latent space, we impose clearer geometric organization between the two distributions using the fine-tuned degradation VAE, illustrated in the right side of Figure 1 (a), simplifying the transport dynamics and improving training stability. While latent space recovery exhibits distortion when decoding to image space, we introduce a VAE-based high-frequency refinement module to mitigate spatial distortion.

Building upon these components, we propose **SLAIR**, a structured latent flow matching for all-in-one image restoration, which learns a deterministic ODE that maps degraded latent to their clean counterparts through continuous transformation as presented in Figure 1 (b). SLAIR provides a unified and stable alternative to regression-based and diffusion-based restoration approaches, achieving high perceptual fidelity while remaining computationally efficient.

Our contributions are summarized as follows:

- a transport-based formulation for all-in-one image restoration, eliminating explicit degradation modeling and unifying diverse tasks;
- a latent-space structuring strategy that enlarges the separation between degraded and clean latents;
- a high-frequency details refinement module that reduces distortions.

We evaluate SLAIR on 9 datasets including 8 tasks, which demonstrates the best overall performance on degradation removal and detail recovery.

2 Related Work

2.1 Diffusion Models for Image Restoration

Image restoration is a prominent application of diffusion models, due to their powerful generative priors. Following DDPM [16], works such as DiffIR [56] and ResShift [63] utilize ϵ -prediction or x_0 -prediction to refine degraded inputs. More recently, Flow Matching (FM) [26, 27] (also called v -prediction [48]) has emerged as an efficient alternative, offering straight vector field paths that reduce inference latency and improve training stability. Building on this paradigm, several works model restoration as direct transport between degraded and clean distributions: PnP-Flow [36] and ResFlow [43] learn deterministic flow paths, PMRF [39] bridges degraded distributions to MMSE reconstructions before transporting to clean, and ELIR [6] performs latent flow matching for blind face restoration.

2.2 All-in-One Image Restoration

All-in-one image restoration aims to handle multiple degradations within a single unified model. Early works [12, 24] often adopt multi-head architectures that require an explicit assessment of degradation types to select specific branches. To achieve blind restoration, AirNet [22] utilizes contrastive learning for degradation representation, while IDR [66] reformulates degradations into fundamental physical ingredients. To enable unified handling of multiple tasks, prompt-based

methods [32, 41, 52] introduce task-specific information via tunable prompts, while other approaches [17, 31, 65] obtain degradation priors by training separate classifiers or leveraging multimodal priors derived from Vision-Language Models (VLM). Beyond spatial priors, AdaIR [8] explores the frequency domain to capture diverse degradation characteristics via adaptive spectral decomposition and modulation. Meanwhile, from a structural perspective, MoE-based [57, 61, 64, 70] frameworks leverage dynamic routing mechanisms to adaptively partition the parameter space. Nevertheless, these methods are typically characterized by high computational costs and a deficiency in generative priors, which limits their capability to reconstruct high-fidelity details under extreme degradations.

2.3 Diffusion-based All-in-One Image Restoration

Due to their superior generative priors, diffusion models have increasingly become a dominant paradigm for all-in-one image restoration. Existing methods [3, 7, 19, 25, 56, 71] leverage diffusion models with a U-Net backbone for multi-task restoration; however, they are often constrained by limited receptive fields and struggle to balance computational efficiency with restoration precision. Recently, Diffusion Transformers (DiT) have demonstrated superior scalability and modeling capacity. Some methods [4, 10] have begun to adapt DiT to image restoration, achieving promising performance. In parallel, while transport-based methods [6, 36, 39, 43] provide a deterministic and efficient generative paradigm, they remain confined to single-source per-task settings. Extending them to all-in-one is ill-posed without explicit latent structuring, as multiple degraded distributions must converge to a single shared clean target. Notably, LightningDiT [60] integrates flow matching within the DiT framework and aligns the latent space with vision foundation models, achieving state-of-the-art performance with accelerated convergence. Inspired by this, we adopt a DiT-based architecture to unify all-in-one restoration within a deterministic flow framework.

3 Method

As shown in Figure 2, given a degraded image I_{deg} and its ground-truth (GT) image I_{gt} , our goal is to learn a unified restoration system that is (i) degradation-agnostic, (ii) perceptually faithful, and (iii) practical across resolutions. We work in a latent space defined by a VAE with encoder $E(\cdot)$ and decoder $D(\cdot)$:

$$z_{\text{deg}} = E(I_{\text{deg}}), \quad z_{\text{gt}} = E(I_{\text{gt}}), \quad \hat{I}_{\text{deg}} = D(z_{\text{deg}}), \quad \hat{I}_{\text{gt}} = D(z_{\text{gt}}), \quad (1)$$

where $\hat{I}_{\text{deg}}$ and $\hat{I}_{\text{gt}}$ are reconstructed images from VAE.

Our framework, SLAIR, is built on three design principles: (1) latent separation before transport, ensuring reliable transport by structuring the latent space; therefore, we update the VAE to explicitly separate degraded and clean latents while maintaining reconstruction ability as described in Section 3.2; (2) deterministic continuous transport, where we learn a conditional vector field to

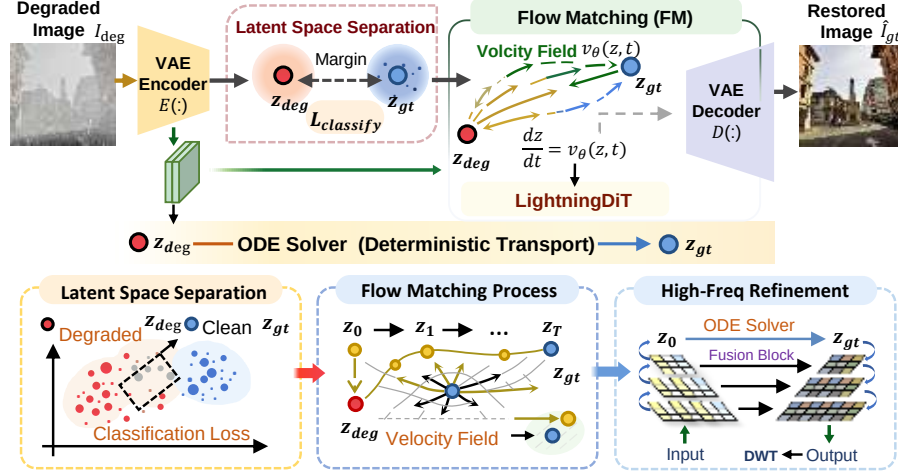


Fig. 2: Overview of the SLAIR framework. A degraded image is first encoded into latent space via VAE in Section 3.2, where a classification-based latent separation loss is adopted to encourage a clear distinction between degraded z_{deg} and clean latents z_{gt} . Flow matching then models deterministic transport from z_{deg} to z_{gt} through a learned velocity field solved by an ODE solver with a LightningDiT [60] backbone as described in Section 3.3. Finally, a high-frequency refinement module proposed in Section 3.4 further enhances details via wavelet-based fusion, producing the restored image.

map $z_{deg} \rightarrow z_{gt}$ without degradation classification; accordingly, we learn the transport vector field in latent space via flow matching and perform inference by solving an ODE in Section 3.3; (3) high-frequency details refinement with the VAE decoder in Section 3.4, which avoids geometric distortion during inference; thus, we achieve distortion-free inference through a VAE refinement module and positional encoding interpolation.

3.1 Preliminaries

Diffusion Models and Latent Diffusion. SLAIR is based on latent diffusion and adopts the flow matching strategy to learn the flow transport from degradation latents to clean latents. Diffusion models (DMs) learn data distributions by progressively corrupting a clean image into noise and training a neural network to reverse this process. However, directly operating in pixel space is computationally expensive, especially at high resolutions due to quadratic attention costs. To address this, Latent Diffusion Models (LDMs) introduce a variational autoencoder (VAE) to map images into a lower-dimensional latent space, significantly reducing spatial complexity [47]. The encoder $E(\cdot)$ projects an input image I into a latent representation z , while the decoder $D(\cdot)$ reconstructs the image from the latent code, *i.e.*, $z = E(I)$ and $\hat{I} = D(z)$. The denoising models are processed in the latent space.

Flow Matching. Diffusion models use denoising-based approaches for generation by reversing a noising process. The model is trained to recover clean samples from noisy observations through a denoising process. However, Flow Matching offers an alternative approach based on continuous-time transport. In FM, instead of denoising, we learn a neural vector field $v_\theta(x, t)$ that directly transports samples from a source distribution to a target distribution through an ODE. Compared to denoising, FM provides the advantage of learning deterministic transport dynamics without the need for stochastic sampling or noise prediction, enabling more stable and efficient mappings from degraded to clean data.

3.2 Degradation-Aware Latent Separation

As shown in Figure 2, our transport model aims to learn a vector field that moves a degraded latent z_{deg} toward its clean counterpart z_{gt} . As shown in Figure 1 (a), different degradations have obvious small clusters, which indicates that VAE has potential for degradation identification. However, the clean z_{gt} and degradation z_{deg} are *entangled* in latent space. Therefore, using the latent space defined by the original VAE raises two issues: **(i) ambiguous transport direction:** small and inconsistent displacements make the target velocity ill-conditioned under noise and data variability; **(ii) unstable training:** the learned vector field must vary sharply in local neighborhoods to separate overlapped manifolds, thereby increasing optimization difficulty and harming generalization. To alleviate these issues, we explicitly structure the latent space within VAE to enforce separability between degraded and clean representations, thereby rendering the subsequent transport problem better posed. Specifically, this is achieved by augmenting the original VAE objective [60] with a separation loss that enforces a margin between the latent representations of degraded and clean images.

Classification-Guided Separation. To structure the latent space, we introduce a classification-guided separation mechanism. Specifically, we append a lightweight auxiliary classifier $f_{\text{classify}}(\cdot)$, typically implemented as a multi-layer perceptron (MLP), to the encoder. For a given latent representation z , the classifier performs a two-level task: a binary domain classification to distinguish between clean and degraded representations, and a fine-grained classification to identify specific degradation types. The total classification objective $\mathcal{L}_{\text{classify}}$ is formulated as a weighted sum of two cross-entropy losses:

$$\mathcal{L}_{\text{classify}} = \lambda_{\text{binary}} \mathcal{L}_{\text{CE}}(\hat{\mathbf{p}}_{\text{binary}}, y_{\text{binary}}) + \lambda_{\text{sub}} \mathcal{L}_{\text{CE}}(\hat{\mathbf{p}}_{\text{sub}}, y_{\text{sub}}), \quad (2)$$

where $\hat{\mathbf{p}}_{\text{binary}}$ and $\hat{\mathbf{p}}_{\text{sub}}$ are derived from the classifier output $\hat{\mathbf{p}} = f_{\text{cls}}(z)$. Specifically, $\hat{\mathbf{p}}_{\text{binary}}$ is the probability distribution over the clean and degraded domains; $\hat{\mathbf{p}}_{\text{sub}}$ represents the fine-grained probability distribution across specific degradation types. $y_{\text{binary}} \in \{0, 1\}$ denotes the GT domain label (where 1 represents the clean domain and 0 represents the degraded domain), while y_{sub} is the GT label for the specific degradation category. Specifically, the loss weights are empirically set to $\lambda_{\text{binary}} = 0.015$ and $\lambda_{\text{sub}} = 0.01$ to balance the fundamental

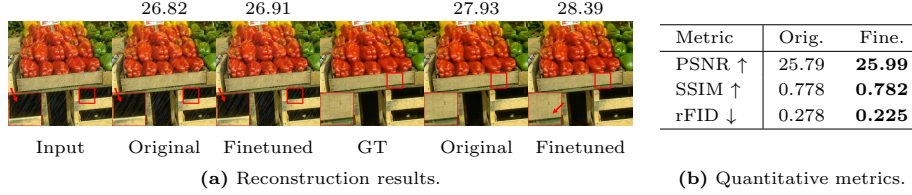


Fig. 3: Comparison between original VAE [60] and our finetuned one. (a) The finetuned VAE can better capture high-frequency details both for degradation and ground truth. The numbers above the images denote PSNR values. (b) Although the classification separation is introduced, the reconstruction ability is still preserved and even better. The reported metrics are calculated on ImageNet [9].

reconstruction fidelity with the newly introduced latent structuring constraints. Here, the labels are introduced solely to facilitate latent space disentanglement and do not constrain the range of degradations the model can address, as the actual degradation removal is carried out within the diffusion process.

VAE loss. Original VAE [60] is trained with reconstruction and KL loss:

$$\mathcal{L}_{\text{vae}} = \|I_{\text{gt}} - D(z_{\text{gt}})\|_1 + \lambda_p \text{LPIPS}(I_{\text{gt}}, D(z_{\text{gt}})) + \lambda_{\text{kl}} \text{KL}(q(z|I_{\text{gt}}) \| p(z)). \quad (3)$$

The separation constraint increases the inter-manifold gap between degraded and clean representations while preserving reconstruction fidelity. This has two direct benefits for transport learning: **(1) stable supervision:** the displacement $u = z_{\text{gt}} - z_{\text{deg}}$ becomes sufficiently informative and less sensitive to local noise; **(2) smoother dynamics:** with reduced manifold overlap, the vector field does not need to “untangle” representations locally, leading to more stable optimization and better generalization. Therefore, our final latent-structuring loss is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{vae}} + \mathcal{L}_{\text{classify}}. \quad (4)$$

We only modify the training objective and the VAE architecture remains unchanged. To qualitatively verify the effectiveness of the proposed representation separation, we employ t-SNE to visualize the latent distributions. The separation loss encourages a clear margin between the two distributions as shown in Figure 1 (a), preventing overlap between degraded and clean representations and ensuring that the flow dynamics exhibit a well-defined direction and magnitude. We achieve higher reconstruction quality as illustrated in Figure 3, indicating that this latent space separation does not influence the reconstruction abilities.

3.3 Conditional Transport via Flow Matching

In image restoration, many learning-based methods estimate the degradation type [19, 22, 40] or degradation-related features [1, 31], while some approaches

rely on stochastic iterative sampling procedures [30, 31]. However, these approaches come with inherent limitations: **(1) degradation classification:** incorrect degradation labels or features can lead to errors in restoration; and **(2) stochastic sampling:** random initial noise introduces instability and inefficiencies in the restoration process. To address these challenges, we propose an alternative way illustrated in Figure 2: modeling restoration as a deterministic flow transformation from the degraded latent representation z_{deg} to the clean latent representation z_{gt} . Unlike methods that require explicit degradation classification, our approach directly learns the conditional flow between z_{deg} and z_{gt} through a vector field by FM, which is easy to scale up to an unlimited number of degradations. FM can avoid error propagation and provide a stable alternative to stochastic sampling, while retaining the expressive power of generative modeling through a learned vector field.

Path Construction under Paired Supervision. With paired degraded latent z_{deg} and target latent z_{gt} , we construct intermediate states of the sample step t along a simple interpolation path:

$$x_t = (1 - t)z_{\text{deg}} + tz_{\text{gt}}, \quad t \sim \mathcal{U}(0, 1), \quad (5)$$

whose time derivative provides a natural supervision signal:

$$u_t = \frac{dx_t}{dt} = z_{\text{gt}} - z_{\text{deg}}. \quad (6)$$

This choice turns restoration into learning a *conditional velocity field* that consistently points from degraded representations to clean ones.

Bridge Noise [72] for Robustness. Training only on the noiseless path can lead to a brittle vector field that overfits to a thin trajectory manifold. To encourage local consistency and improve robustness, we perturb intermediate states with bridge noise following DDBM [72]:

$$\tilde{x}_t = x_t + \sqrt{t(1-t)}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \quad (7)$$

The factor $\sqrt{t(1-t)}$ injects stronger perturbations near the middle of the path while keeping endpoints stable, which empirically stabilizes training and improves generalization.

Conditional Velocity Regression. We parameterize the vector field with a conditional DiT backbone $v_\theta(\cdot)$ based on LightningDiT [60]:

$$\mathcal{L}_{\text{fm}} = \mathbb{E} [\|v_\theta(\tilde{x}_t, t; z_{\text{deg}}) - u_t\|_2^2]. \quad (8)$$

We add a cosine alignment term to emphasize directional correctness:

$$\mathcal{L}_{\text{cos}} = \mathbb{E} [1 - \cos(v_\theta(\tilde{x}_t, t; \cdot), u_t)], \quad (9)$$

resulting in the final transport loss:

$$\mathcal{L}_{\text{transport}} = \mathcal{L}_{\text{fm}} + \mathcal{L}_{\text{cos}}. \quad (10)$$

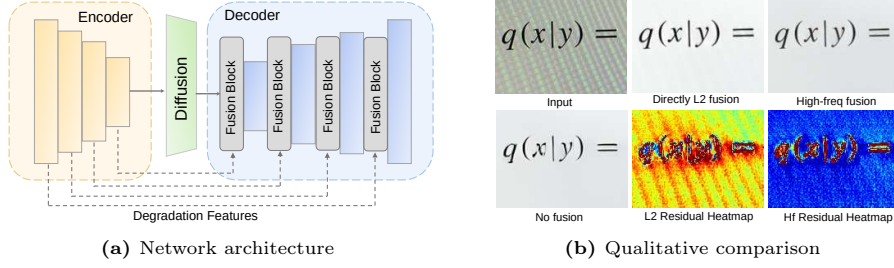


Fig. 4: Illustration of the high-frequency details refinement. (a) Multi-scale degradation features are integrated into the VAE decoder via fusion blocks. (b) Visual comparison of fusion losses. Residual heatmaps are obtained by subtracting the “No fusion” output. Compared to “Directly L2 fusion” (which reintroduces degradation), our proposed “High-freq fusion” effectively recovers sharp edges while keeping the background clean.

Inference. At inference time, we start from $\hat{z}(0) = z_{\text{deg}}$ and solve:

$$\frac{d\hat{z}(t)}{dt} = v_{\theta}(\hat{z}(t), t; z_{\text{deg}}), \quad t : 0 \rightarrow 1, \quad (11)$$

to obtain $\hat{z}(1)$, which is then decoded into the restored image. This formulation decouples restoration from stochastic sampling and yields stable predictions under identical inputs. By learning this deterministic transport, our method benefits from the expressive power of generative modeling while providing a stable and efficient restoration process.

3.4 High-frequency Details Refinement

Although performing in latent space reduces computational complexity, it can result in the loss of high-frequency details. To mitigate this issue, we adopt high-frequency details refinement with the help of degradation features as illustrated in Figure 2. We introduce pyramid VAE encoder features to the decoding process as shown in Figure 4 (a), ensuring that the high-frequency information is introduced when transitioning from latent space to pixel space by adopting L2 loss and perceptual loss as below:

$$\mathcal{L}_{\text{mse}} = \mathbb{E}[\|I_{\text{gt}} - \hat{I}\|_2^2], \quad \mathcal{L}_{\text{perc}} = \mathbb{E}[\|\phi(I_{\text{gt}}) - \phi(\hat{I})\|_2^2], \quad (12)$$

where I_{gt} and $\hat{I}$ denote the GT image and the reconstructed image, respectively; $\phi(\cdot)$ represents the feature extraction operator of a pre-trained perceptual network VGG [49]. However, as shown in Figure 4 (b), when performing direct regression using Equation (12), degradation may be reintroduced during decoding, even if it has been removed in the latent space. This happens because the decoder, when introducing high-frequency features from degradation inputs, might inadvertently introduce the high-frequency degradation which is entangled with high-frequency details, thereby introducing unavoidable degradation especially for high-frequency degradation tasks (*e.g.*, deraining, demoiréing).

To avoid introducing high-frequency degradation from the inputs, we introduce a frequency-based refinement loss, which optimizes the decoding process by incorporating non-degradation details from the degraded image, thus preventing the reintroduction of degradation information. Specifically, we employ a Wavelet-based high-frequency loss $\mathcal{L}_{\text{hf}}$, defined as:

$$\mathcal{L}_{\text{hf}} = \|\text{DWT}_{\text{hf}}(I_{\text{gt}}) - \text{DWT}_{\text{hf}}(\hat{I})\|_1, \quad (13)$$

where $\text{DWT}_{\text{hf}}(\cdot)$ represents the Discrete Wavelet Transform (DWT) that extracts the horizontal, vertical, and diagonal high-frequency coefficients. By minimizing the L_1 distance in the wavelet domain, we explicitly force the decoder to recover sharp edges and fine textures, thereby suppressing the reintroduction of degradation artifacts as illustrated in Figure 4 (b).

3.5 Implementation Details

Latent Separation. VAE for latent separation follows the original architecture of LightningDiT [60] and is implemented using PyTorch. Training is conducted at a resolution of 256×256 with a batch size of 16 in a full fine-tuning manner. Adam optimizer with learning rate 3×10^{-6} is applied, optimized for 1 epoch on a single NVIDIA RTX 5090 GPU. Furthermore, to preserve the original reconstruction capability of the VAE, we employ a combination of the aforementioned task-specific data and a diverse portion of the ImageNet [9] training set.

Transport ODE. Our transport ODE is implemented using PyTorch, and trained with a resolution of 512×512 and batch size of 12. We train our ODE backbone from scratch with AdamW optimizer and learning rate 10^{-4} , $\beta = (0.9, 0.95)$ for 100k steps on two NVIDIA RTX 5090 GPUs.

Details Refinement. We focus on optimizing the fusion layers within the VAE decoder to adaptively integrate multi-scale degradation priors. During this stage, the VAE backbone and transport model remain frozen, while only the fusion modules are updated using the Adam optimizer, a batch size of 2, and a learning rate of 1×10^{-4} for 1 epoch on a single NVIDIA RTX 5090 GPU.

4 Experiments

We evaluate SLAIR under two settings. *(i) Comprehensive 8-task:* we evaluate on 8 degradation types across 9 datasets, where compared methods are evaluated using their official pretrained models. *(ii) Standard all-in-one:* following previous works [8, 41, 64], we train SLAIR on the standard 5-task setting (denoising, deraining, deblurring, dehazing, and low-light enhancement). Additional experiments, including ablation studies and analysis on generalization ability, are provided in the supplementary material.

Table 1: Quantitative comparison on all benchmarks under setting (i). **Bold** and underline indicate the best and the second best performance, respectively.

Method	Rain1400				Outdoor-Rain				Raindrop				UHDSnow				SOTS			
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$
AirNet [22]	24.32	.764	.207	85.49	16.39	.600	.457	200.99	21.91	.832	.191	119.98	22.14	.780	.232	106.28	26.57	.925	.056	19.97
PromptIR [41]	25.40	.798	.176	73.90	<u>17.22</u>	<u>.614</u>	.445	190.44	22.79	.836	.174	119.66	23.12	.847	.177	82.21	28.90	.974	.016	<u>4.26</u>
DA-CLIP [31]	25.89	.816	.158	<u>63.91</u>	15.15	.544	.483	191.80	29.74	<u>.822</u>	<u>.052</u>	34.25	31.43	<u>.944</u>	<u>.042</u>	18.76	<u>30.94</u>	.965	.016	4.67
DiffUIR [71]	30.19	.903	.072	25.52	14.74	.603	<u>.431</u>	<u>132.02</u>	23.50	.844	.170	113.43	<u>29.41</u>	<u>.943</u>	<u>.056</u>	<u>27.59</u>	31.11	.976	.011	9.53
AdaIR [8]	24.52	.760	.205	83.96	16.96	.611	.445	191.35	23.00	.840	.174	121.01	22.47	.792	.222	99.96	27.56	.939	.041	11.36
DCPT [17]	25.25	.792	.184	79.31	13.51	.554	.501	199.03	18.28	.788	.199	118.58	22.95	.837	.189	89.27	29.58	.973	.016	4.44
MoCE-IR [64]	24.70	.773	.203	87.27	16.66	.610	.449	191.79	22.40	.829	.178	117.44	22.56	.812	.217	101.85	30.08	<u>.975</u>	<u>.014</u>	4.22
VLU-Net [65]	24.55	.765	.201	84.64	17.02	.612	.446	190.74	22.88	.837	.179	115.29	22.38	.784	.229	103.67	27.68	.941	.043	12.44
Ours	<u>29.36</u>	<u>.893</u>	<u>.080</u>	<u>29.64</u>	22.50	.815	.187	55.75	<u>28.74</u>	.904	<u>.082</u>	<u>45.66</u>	28.62	.925	.068	30.54	26.02	.938	.034	8.89

Method	BSD68				LoL				UHDM				HDR-Eye				Average			
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$
AirNet [22]	26.17	.636	.429	74.70	24.01	.720	.278	97.91	18.22	.698	.234	54.89	16.04	.753	.175	62.10	21.75	.745	.251	91.37
PromptIR [41]	22.32	.445	.564	83.77	25.23	.771	.228	97.15	18.57	.703	.223	53.04	16.18	.740	.172	55.49	22.19	.748	.242	84.44
DA-CLIP [31]	<u>27.98</u>	<u>.727</u>	<u>.192</u>	<u>60.57</u>	26.57	.804	.210	124.59	18.61	.727	.222	54.06	16.65	.785	.162	58.48	<u>24.77</u>	<u>.806</u>	<u>.171</u>	67.90
DiffUIR [71]	20.41	.558	.473	86.91	19.35	.809	.180	<u>89.85</u>	18.57	<u>.737</u>	<u>.208</u>	<u>48.62</u>	<u>17.41</u>	<u>.815</u>	<u>.149</u>	54.28	22.74	.799	.194	65.31
AdaIR [8]	23.04	.478	.546	83.58	22.14	.734	.256	102.82	18.76	.719	.222	52.09	17.34	.784	.165	56.88	21.75	.740	.253	89.22
DCPT [17]	22.38	.457	.566	82.82	26.02	.789	.212	100.60	17.81	.715	.230	54.91	13.96	.799	.186	64.68	21.08	.745	.254	88.18
MoCE-IR [64]	25.31	.587	.456	76.79	27.50	.836	<u>.169</u>	96.04	18.54	.708	.226	53.47	16.25	.758	.172	56.03	22.67	.765	.232	87.21
VLU-Net [65]	23.45	.495	.516	81.66	21.09	.820	.175	97.94	18.71	.719	.222	52.48	16.44	.750	.171	55.43	21.58	.747	.242	88.25
Ours	28.36	.858	.178	48.78	<u>26.83</u>	<u>.833</u>	.139	69.76	21.59	.847	.114	26.45	23.29	.870	.148	<u>54.36</u>	26.15	.876	.114	41.09

Datasets. Setting (i): We prepare datasets covering 9 benchmarks across 8 degradation types. 1) Denoising: we utilize a combined set of the BSD400 [2] and WED [33] datasets for training, which consists of 400 and 4,744 samples, respectively, and evaluate on BSD68 [35] with 68 images. 2) Deraining: we employ the Rain1400 [11] dataset which contains a total of 12,600 image pairs, and evaluate on its test split of 1,400 images. 3) Deraining & Dehazing: we evaluate on Outdoor-Rain [23], which combines rain, haze, and low-light degradations, including 750 images. 4) Raindrop removal: we combine the RainDrop [42], RaindropClarity [20], and RainDS [45] datasets, providing a joint training set of 10,911 samples, and evaluate on RainDrop with 58 images. 5) Desnowing: we utilize the Snow100K [29] dataset for training, which comprises 50,000 images, and evaluate on UHDSnow [53] with 200 images. 6) Dehazing: we utilize SOTS [21] that contains 72,135 training images, and evaluate on its test split of 500 images. 7) Demoiréing: we combine the UHDM [62] and FHDMi [13] datasets, resulting in a total of 14,481 training pairs, and evaluate on UHDM with 500 images. 8) Low-light enhancement: we utilize the LoL-v1 [54] dataset, which contains 485 image pairs, and evaluate on its test split of 15 images. 9) High Dynamic Range (HDR) reconstruction: we synthesize over-exposed and under-exposed images using data from HDR-SYNTH [28] and HDR-REAL [28]; after a filtering process, we construct a comprehensive training set comprising 9,438 over-exposed and 7,763 under-exposed image pairs, and evaluate on HDR-Eye [38]. The proposed model is trained on the joint set of the aforementioned training data. All evaluations are performed at a resolution of 512×512 . **Setting (ii):** Following prior works [8, 41, 64], we train on the standard 5-task benchmark. For image denoising, we combine BSD400 [2] and WED [33], adding Gaussian noise at levels $\sigma \in \{15, 25, 50\}$, and evaluate on BSD68 [35]. For deraining, deblurring, and dehazing, we follow the standard setting, using Rain100L [58], GoPro [37], and SOTS [21] for both training and evaluation, respectively. For low-light enhancement, we utilize LoL-v1 [54] for both training and evaluation.

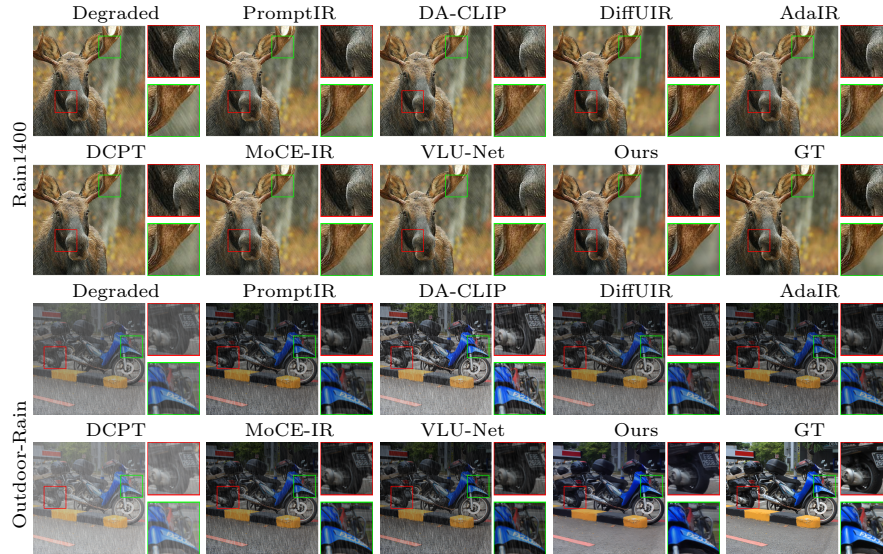


Fig. 5: Qualitative comparison on Rain1400 [11] and Outdoor-Rain [23]. Compared with existing methods, our SLAIR produces better deraining results and finer details.

Methods. To evaluate the performance, we compare the proposed method with a diverse set of representative baselines, including CNN-based discriminative models and recent diffusion-based generative approaches: 1) AirNet [22]: extracts degradation-related information via contrastive learning that is used for guiding subsequent restoration; 2) PromptIR [41]: encodes degradation information into the pre-set prompts; 3) DA-CLIP [31]: leverages pre-trained vision-language models via image and text encoders; 4) DiffUIR [71]: maps different degradations into a shared distribution via diffusion-based selective hourglass mapping. 5) AdaIR [8]: mines and modulates the frequency priors to enhance image restoration; 6) DCPT [17]: expands and utilizes the degradation classification ability of the restoration model; 7) MoCE-IR [64]: leverages the MoE layer to selectively activate complexity experts based on input requirements; 8) VLU-Net [65]: integrates an automatic gradient estimation strategy based on vision-language models for multiple degradation restoration.

Quantitative Evaluation. We adopt four metrics for quantitative evaluation: peak signal-to-noise ratio (PSNR), structural similarity (SSIM), the perceptual error with learned perceptual image patch similarity (LPIPS) [69], and Fréchet Inception Distance (FID) [15]. **Setting (i).** As shown in Table 1, SLAIR achieves state-of-the-art average performance across all datasets, improving PSNR by 5.6% and reducing LPIPS by 33% over top baselines. It demonstrates remarkable performance, with PSNR gains on UHDM (15%) and Outdoor-Rain (30%), alongside improvements on BSD68 and HDR-Eye. While some baselines remain

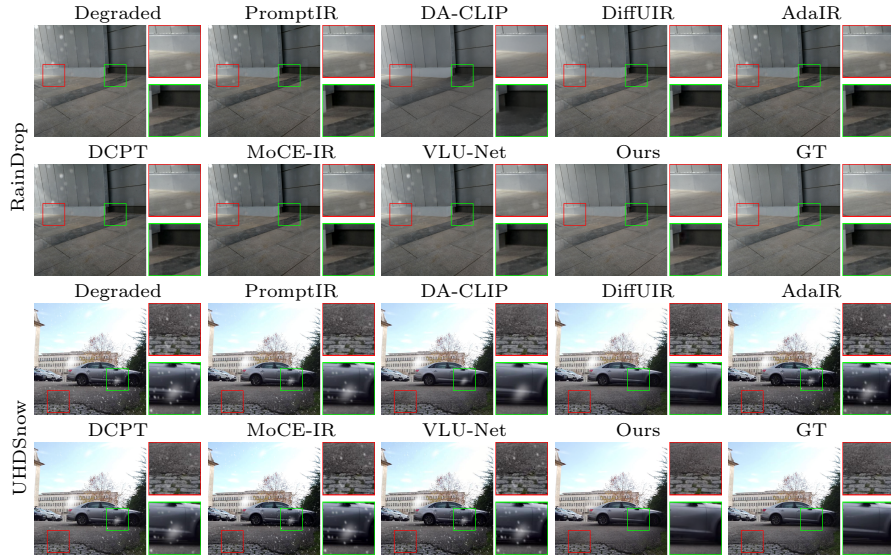


Fig. 6: Qualitative comparison on RainDrop [42] and UHDSnow [53]. Our SLAIR exhibits better raindrop and snow removal ability.

competitive on individual weather benchmarks such as Rain1400, Raindrop, and UHDSnow, SLAIR exhibits strong overall robustness across diverse degradation types. **Setting (ii).** As shown in Table 2, under the standard 5-task benchmark, SLAIR achieves the best SSIM, LPIPS, and FID among all compared methods, including the diffusion-based DiffUIR [71], demonstrating competitive performance under a fair and aligned training setting.

Qualitative Evaluation. Qualitative comparison for Setting (i) is depicted in Figures 5 to 7 with one sample per dataset. Overall, our method produces visually more faithful results with sharper structures and fewer artifacts. For weather-related degradations (Rain1400, Outdoor-Rain, RainDrop, and UHDSnow), existing methods may leave noticeable rain streaks or over-smooth image details, while our approach effectively removes degradations while preserving fine textures, such as the fur details of the deer and the sharp edges of the motorcycle. On RainDrop and UHDSnow, our method better recovers structural details and avoids blurring artifacts observed in several baselines. For other restoration tasks, including BSD68, LoL, UHDM, and HDR-Eye, our method also produces results that are visually closer to the ground truth, with improved texture fidelity and more natural brightness and contrast. These visual comparisons highlight the effectiveness and strong generalization ability of the proposed framework across diverse restoration scenarios. Qualitative comparisons for Setting (ii) are included in the supplementary material, further demonstrating that SLAIR maintains consistent visual advantages under the standard setting.

Table 2: Performance on the standard 5-task all-in-one benchmark (BSD68, GoPro, LoL, Rain100L, SOTS). **Bold** and underline indicate the best and the second best performance, respectively.

Method	BSD68				GoPro				LoL			
	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
AirNet [22]	26.17	.636	.429	74.70	22.85	.766	.275	60.57	24.01	.720	.278	97.91
PromptIR [41]	22.32	.445	.564	83.77	22.65	.763	.276	59.29	25.23	.771	.228	97.15
DiffUIR [71]	20.41	.558	.473	86.91	27.62	.844	.201	40.99	19.35	.809	.180	<u>89.85</u>
AdaIR [8]	23.04	.478	.546	83.58	<u>28.98</u>	<u>.878</u>	.161	<u>28.23</u>	22.14	.734	.256	102.82
DCPT [17]	22.38	.457	.566	82.82	22.20	.799	.242	45.09	26.02	.789	.212	100.60
MoCE-IR [64]	25.31	.587	.456	76.79	30.46	.907	.125	22.36	<u>27.50</u>	<u>.836</u>	<u>.169</u>	96.04
VLU-Net [65]	23.45	.495	.516	81.66	28.25	.860	.184	34.67	21.09	.820	.175	97.94
Ours	29.21	.846	.139	39.99	25.91	.796	<u>.153</u>	32.87	29.13	.883	.083	45.75

Method	Rain100L				SOTS				Average			
	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
AirNet [22]	28.76	.881	.147	41.40	26.57	.925	.056	19.97	25.67	.786	.237	58.91
PromptIR [41]	32.16	<u>.902</u>	<u>.105</u>	26.64	28.90	.974	.016	<u>4.26</u>	26.25	.771	.238	54.22
DiffUIR [71]	28.26	.898	.165	79.52	31.11	.976	.011	9.53	25.35	.817	.206	61.36
AdaIR [8]	28.85	.858	.177	70.13	27.56	.939	.041	11.36	26.11	.777	.236	59.22
DCPT [17]	30.61	.899	.119	42.12	29.58	.973	.016	4.44	26.16	.783	.231	55.01
MoCE-IR [64]	29.26	.863	.160	61.48	<u>30.08</u>	<u>.975</u>	<u>.014</u>	4.22	28.52	<u>.834</u>	<u>.185</u>	<u>52.18</u>
VLU-Net [65]	31.70	.897	.115	34.55	27.68	.941	.043	12.44	26.43	.803	.207	52.25
Ours	31.01	.943	.060	<u>28.89</u>	26.67	.942	.033	7.76	<u>28.39</u>	.882	.094	31.05

5 Conclusion

We proposed SLAIR, a structured latent flow matching framework for all-in-one image restoration. By formulating restoration as a deterministic transport within an separated latent space, SLAIR bypasses degradation estimation and stochastic sampling. To preserve fine details during decoding, we introduced a wavelet-based high-frequency refinement to suppress artifacts. Experiments across 8 diverse tasks demonstrate the state-of-the-art performance of SLAIR.

Limitations. While verifying the feasibility of latent-space flow matching, our purely degradation-agnostic design trades off controllability; explicitly incorporating degradation labels or prompts could enable interactive adjustments, such as controlling the restoration strength. Besides, constrained by computational resources and the limited scale of high-resolution datasets, our training is restricted to 512×512 patches, although SLAIR supports higher resolution.

Acknowledgements

This work was supported by National Key R&D Program of China (Grant No. 2025YFA1016700), National Natural Science Foundation of China (Grant No. 62136001), Beijing Natural Science Foundation (Grant No. L233024), Beijing Municipal Science & Technology Commission, Administrative Commission of Zhongguancun Science Park (Grant No. Z241100003524012), and Beijing High Innovation Plan (Contract No. 202604841443).

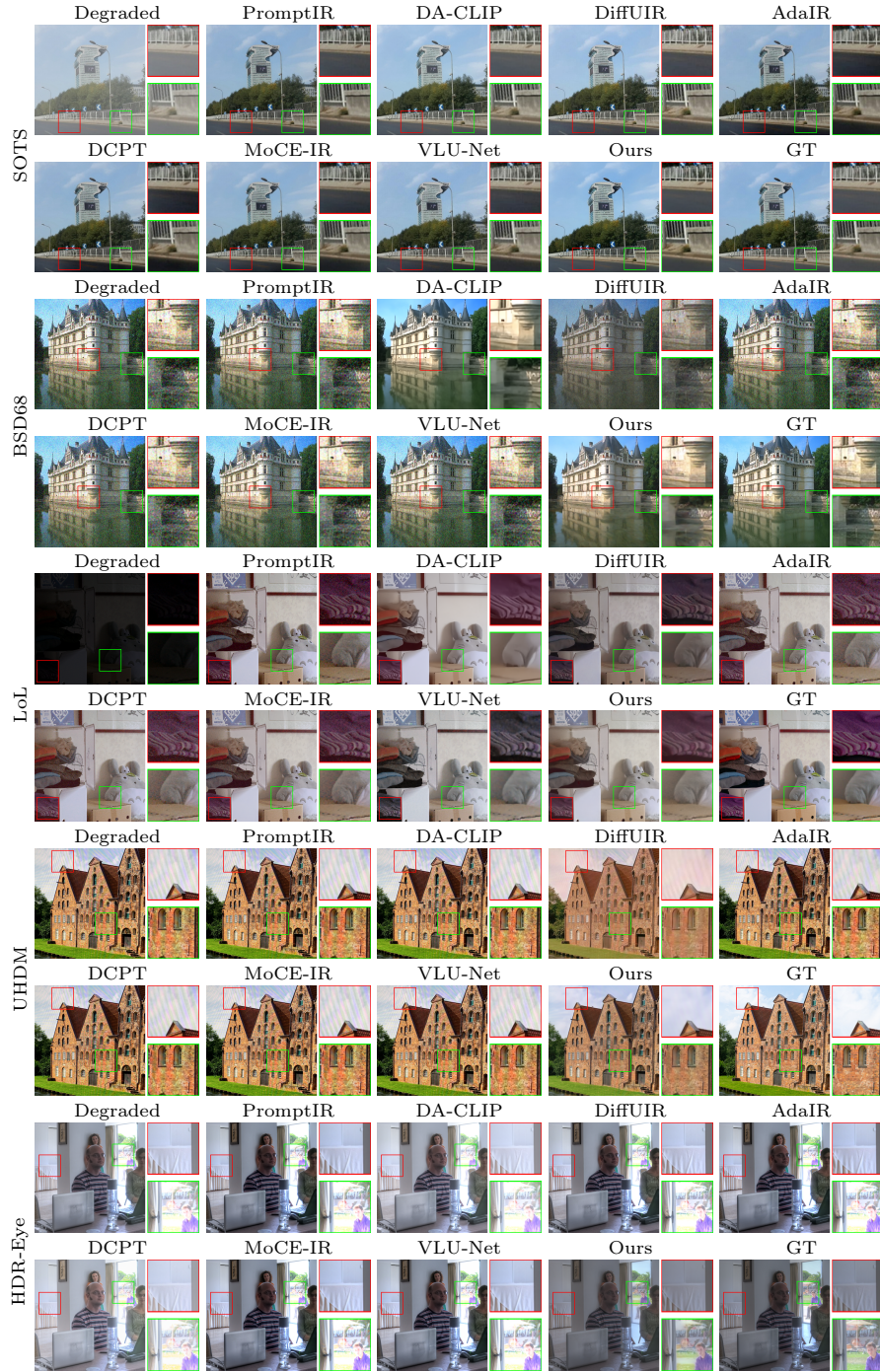


Fig. 7: Qualitative comparison on SOTS [21], BSD68 [35], LoL [54], UHDM [62] and HDR-Eye [38]. The proposed SLAIR (“Ours”) exhibits better restoration ability.

References

1. Ai, Y., Huang, H., Zhou, X., Wang, J., He, R.: Multimodal Prompt Perceiver: Empower adaptiveness generalizability and fidelity for all-in-one image restoration. In: IEEE Conf. Comput. Vis. Pattern Recog. (2024)
2. Arbeláez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* (2011)
3. Chen, I.H., Chen, W.T., Liu, Y.W., Chiang, Y.C., Kuo, S.Y., Yang, M.H.: UniRestore: Unified perceptual and task-oriented image restoration model using diffusion prior. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
4. Cheng, K., Yu, L., Tu, Z., He, X., Chen, L., Guo, Y., Zhu, M., Wang, N., Gao, X., Hu, J.: Effective diffusion transformer architecture for image super-resolution. In: AAAI (2025)
5. Cheng, Z., Zhou, L., Chen, D., Tang, N., Luo, X., Qu, Y.: Unildiff: Unlocking the power of diffusion priors for all-in-one image restoration. *arXiv preprint arXiv:2507.23685* (2025)
6. Cohen, E., Achituve, I., Diamant, I., Netzer, A., Habi, H.V.: Efficient image restoration via latent consistency flow matching. *arXiv preprint arXiv:2502.03500* (2025)
7. Conde, M.V., Geigle, G., Timofte, R.: InstructIR: High-quality image restoration following human instructions. In: Eur. Conf. Comput. Vis. (2024)
8. Cui, Y., Zamir, S.W., Khan, S., Knoll, A., Shah, M., Khan, F.S.: AdaIR: Adaptive all-in-one image restoration via frequency mining and modulation. In: Int. Conf. Learn. Represent. (2025)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: IEEE Conf. Comput. Vis. Pattern Recog. (2009)
10. Duan, Z.P., Zhang, J., Jin, X., Zhang, Z., Xiong, Z., Zou, D., Ren, J., Guo, C.L., Li, C.: DiT4SR: Taming diffusion transformer for real-world image super-resolution. In: Int. Conf. Comput. Vis. (2025)
11. Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J.: Removing rain from single images via a deep detail network. In: IEEE Conf. Comput. Vis. Pattern Recog. (2017)
12. Han, J., Li, W., Fang, P., Sun, C., Hong, J., Armin, M.A., Petersson, L., Li, H.: Blind image decomposition. In: Eur. Conf. Comput. Vis. (2022)
13. He, B., Wang, C., Shi, B., Duan, L.Y.: FHDe2Net: Full high definition demoireing network. In: Eur. Conf. Comput. Vis. (2020)
14. He, C., Shen, Y., Fang, C., Xiao, F., Tang, L., Zhang, Y., Zuo, W., Guo, Z., Li, X.: Diffusion models in low-level vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* (2025)
15. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Adv. Neural Inform. Process. Syst. (2017)
16. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: Adv. Neural Inform. Process. Syst. (2020)
17. Hu, J., Jin, L., Yao, Z., Lu, Y.: Universal image restoration pre-training via degradation classification. In: Int. Conf. Learn. Represent. (2025)
18. Jiang, J., Zuo, Z., Wu, G., Jiang, K., Liu, X.: A survey on all-in-one image restoration: Taxonomy, evaluation and future trends. *IEEE Trans. Pattern Anal. Mach. Intell.* (2025)
19. Jiang, Y., Zhang, Z., Xue, T., Gu, J.: AutoDIR: Automatic all-in-one image restoration with latent diffusion. In: Eur. Conf. Comput. Vis. (2024)

20. Jin, Y., Li, X., Wang, J., Zhang, Y., Zhang, M.: Raindrop Clarity: A dual-focused dataset for day and night raindrop removal. In: *Eur. Conf. Comput. Vis.* (2024)
21. Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., Wang, Z.: Benchmarking single-image dehazing and beyond. *IEEE Trans. Image Process.* (2017)
22. Li, B., Liu, X., Hu, P., Wu, Z., Lv, J., Peng, X.: All-in-one image restoration for unknown corruption. In: *IEEE Conf. Comput. Vis. Pattern Recog.* (2022)
23. Li, R., Cheong, L.F., Tan, R.T.: Heavy Rain Image Restoration: Integrating physics model and conditional adversarial learning. *IEEE Conf. Comput. Vis. Pattern Recog.* (2019)
24. Li, R., Tan, R.T., Cheong, L.F.: All in one bad weather removal using architectural search. In: *IEEE Conf. Comput. Vis. Pattern Recog.* (2020)
25. Lin, X., He, J., Chen, Z., Lyu, Z., Dai, B., Yu, F., Qiao, Y., Ouyang, W., Dong, C.: DiffBIR: Toward blind image restoration with generative diffusion prior. In: *Eur. Conf. Comput. Vis.* (2024)
26. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: *Int. Conf. Learn. Represent.* (2023)
27. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: *Int. Conf. Learn. Represent.* (2023)
28. Liu, Y.L., Lai, W.S., Chen, Y.S., Kao, Y.L., Yang, M.H., Chuang, Y.Y., Huang, J.B.: Single-image hdr reconstruction by learning to reverse the camera pipeline. *IEEE Conf. Comput. Vis. Pattern Recog.* (2020)
29. Liu, Y.F., Jaw, D.W., Huang, S.C., Hwang, J.N.: DesnowNet: Context-aware deep network for snow removal. *IEEE Trans. Image Process.* (2017)
30. Luo, Z., Gustafsson, F.K., Zhao, Z., Sjölund, J., Schön, T.B.: Image restoration with mean-reverting stochastic differential equations. In: *Int. Conf. Mach. Learn.* (2023)
31. Luo, Z., Gustafsson, F.K., Zhao, Z., Sjölund, J., Schön, T.B.: Controlling vision-language models for universal image restoration. In: *Int. Conf. Learn. Represent.* (2024)
32. Luo, Z., Gustafsson, F.K., Zhao, Z., Sjölund, J., Schön, T.B.: Controlling vision-language models for multi-task image restoration. In: *Int. Conf. Learn. Represent.* (2024)
33. Ma, K., Duanmu, Z., Wu, Q., Wang, Z., Yong, H., Li, H., Zhang, L.: Waterloo exploration database: New challenges for image quality assessment models. *IEEE Trans. Image Process.* (2017)
34. van der Maaten, L., Hinton, G.E.: Visualizing data using t-sne. *Journal of Machine Learning Research* (2008)
35. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: *Int. Conf. Comput. Vis.* (2001)
36. Martin, S., Gagneux, A., Hagemann, P., Steidl, G.: PnP-Flow: Plug-and-play image restoration with flow matching. In: *Int. Conf. Learn. Represent.* (2025)
37. Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. *IEEE Conf. Comput. Vis. Pattern Recog.* (2016)
38. Nemoto, H., Korshunov, P., Hanhart, P., Ebrahimi, T.: Visual attention in ldr and hdr images. In: *9th International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM)* (2015)
39. Ohayon, G., Michaeli, T., Elad, M.: Posterior-mean rectified flow: Towards minimum mse photo-realistic image restoration. In: *ICLR* (2025)

40. Park, D., Lee, B.H., Chun, S.Y.: All-in-one image restoration for unknown degradations using adaptive discriminative filters for specific degradations. *IEEE Conf. Comput. Vis. Pattern Recog.* (2023)
41. Potlapalli, V., Zamir, S.W., Khan, S.H., Shahbaz Khan, F.: PromptIR: Prompting for all-in-one image restoration. In: *Adv. Neural Inform. Process. Syst.* (2023)
42. Qian, R., Tan, R.T., Yang, W., Su, J., Liu, J.: Attentive generative adversarial network for raindrop removal from a single image. *IEEE Conf. Comput. Vis. Pattern Recog.* (2017)
43. Qin, H., Luo, W., Wang, L., Zheng, D., Chen, J., Yang, M., Li, B., Hu, W.: Reversing flow for image restoration. In: *IEEE Conf. Comput. Vis. Pattern Recog.* (2025)
44. Qin, X., Wang, Z., Bai, Y., Xie, X., Jia, H.: FFA-Net: Feature fusion attention network for single image dehazing. In: *AAAI* (2020)
45. Quan, R., Yu, X., Liang, Y., Yang, Y.: Removing raindrops and rain streaks in one go. In: *IEEE Conf. Comput. Vis. Pattern Recog.* (2021)
46. Ren, B., Yang, R., Ma, Q., Zheng, X., Liu, M., Paudel, D.P., Gool, L.V., Cucchiara, R., Sebe, N.: Rethinking expressivity and degradation-awareness in attention for all-in-one blind image restoration. In: *Int. Conf. Learn. Represent.* (2026)
47. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: *IEEE Conf. Comput. Vis. Pattern Recog.* (2022)
48. Salimans, T., Ho, J.: Progressive distillation for fast sampling of diffusion models. In: *Int. Conf. Learn. Represent.* (2022)
49. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014)
50. Sun, S., Ren, W., Wang, T., Cao, X.: Rethinking image restoration for object detection. *Adv. Neural Inform. Process. Syst.* (2022)
51. Tang, X., Gu, X., He, X., Hu, X., Sun, J.: Learning continuous wasserstein barycenter space for generalized all-in-one image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* (2026)
52. Wang, C., Pan, J., Wang, W., Dong, J., Wang, M., Ju, Y., Chen, J.: PromptRestorer: A prompting image restoration method with degradation perception. In: *Adv. Neural Inform. Process. Syst.* (2023)
53. Wang, L., Wang, C., Pan, J., Liu, X., Zhou, W., Sun, X., Wang, W., Su, Z.: Ultra-High-Definition Image Restoration: New benchmarks and a dual interaction prior-driven solution. *IEEE Trans. Circuit Syst. Video Technol.* (2024)
54. Wei, C., Wang, W., Yang, W., Liu, J.: Deep retinex decomposition for low-light enhancement. *ArXiv* (2018)
55. Wu, G., Jiang, J., Jiang, K., Liu, X., Nie, L.: Beyond degradation redundancy: Contrastive prompt learning for all-in-one image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* (2025)
56. Xia, B., Zhang, Y., Wang, S., Wang, Y., Wu, X., Tian, Y., Yang, W., Van Gool, L.: DiffIR: Efficient diffusion model for image restoration. In: *Int. Conf. Comput. Vis.* (2023)
57. Yang, H., Pan, L., Yang, Y., Liang, W.: Language-driven all-in-one adverse weather removal. In: *IEEE Conf. Comput. Vis. Pattern Recog.* (2024)
58. Yang, W., Tan, R.T., Feng, J., Liu, J., Guo, Z., Yan, S.: Deep joint rain detection and removal from a single image. In: *IEEE Conf. Comput. Vis. Pattern Recog.* (2017)
59. Yang, Z., Chen, H., Qian, Z., Yi, Y., Zhang, H., Zhao, D., Wei, B., Xu, Y.: All-in-one medical image restoration via task-adaptive routing. In: *MICCAI* (2024)

60. Yao, J., Yang, B., Wang, X.: Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
61. Yu, X., Zhou, S., Li, H., Zhu, L.: Multi-expert adaptive selection: Task-balancing for all-in-one image restoration. IEEE Trans. Circuit Syst. Video Technol. (2025)
62. Yu, X., Dai, P., Li, W., Ma, L., Shen, J., Li, J., Qi, X.: Towards efficient and scale-robust ultra-high-definition image demoiréing. In: Eur. Conf. Comput. Vis. (2022)
63. Yue, Z., Wang, J., Loy, C.C.: Efficient diffusion model for image restoration by residual shifting. IEEE Trans. Pattern Anal. Mach. Intell. (2024)
64. Zamfir, E., Wu, Z., Mehta, N., Tan, Y., Paudel, D.P., Zhang, Y., Timofte, R.: Complexity experts are task-discriminative learners for any image restoration. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
65. Zeng, H., Wang, X., Chen, Y., Su, J., Liu, J.: Vision-language gradient descent-driven all-in-one deep unfolding networks. IEEE Conf. Comput. Vis. Pattern Recog. (2025)
66. Zhang, J., Huang, J., Yao, M., Yang, Z., Yu, H., Zhou, M., Zhao, F.: Ingredient-oriented multi-degradation learning for image restoration. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
67. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. IEEE Trans. Image Process. (2017)
68. Zhang, K., Zuo, W., Zhang, L.: FFDNet: Toward a fast and flexible solution for cnn-based image denoising. IEEE Trans. Image Process. (2018)
69. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. IEEE Conf. Comput. Vis. Pattern Recog. (2018)
70. Zhang, R., Luo, Y., Liu, J., Yang, H., Dong, Z., Gudovskiy, D., Okuno, T., Nakata, Y., Keutzer, K., Du, Y., Zhang, S.: Efficient deweather mixture-of-experts with uncertainty-aware feature-wise linear modulation. In: AAAI (2024)
71. Zheng, D., Wu, X.M., Yang, S., Zhang, J., Hu, J.F., Zheng, W.S.: Selective hour-glass mapping for universal image restoration based on diffusion model. In: IEEE Conf. Comput. Vis. Pattern Recog. (2024)
72. Zhou, L., Lou, A., Khanna, S., Ermon, S.: Denoising diffusion bridge models. arXiv preprint arXiv:2309.16948 (2023)

Supplementary Material for SLAIR: Structured Latent Flow Matching for All-in-One Image Restoration

Shuyi Liang^{*,1}, Yixin Yang^{*,1}, Hanyue Lou¹,
Yuning Cui², and Boxin Shi^{†1}

¹ Peking University

² Technical University of Munich

liangshuyi@stu.pku.edu.cn yuning.cui@in.tum.de
{yangyixin93, hylz, shiboxin}@pku.edu.cn

In this supplementary material, we provide implementation details and network architecture in Section A, ablation studies in Section B, generalization and adaptation analysis in Section C, High Dynamic Range (HDR) data synthesis in Section D, computational efficiency comparisons in Section E, additional qualitative results in Section F, and the resolution-agnostic inference method with its quantitative and qualitative results in Section G.

A Implementation Details and Network Architecture

A.1 Training Pipeline of Flow-Matching DiT

The detailed training pipeline of our Flow-Matching framework is presented in Algorithm 1. During each training iteration, we sample a time step t to construct the intermediate noisy latent x_t by blending the clean and degraded latents with a small Gaussian perturbation ϵ . The network v_θ is then tasked with predicting the ground-truth vector field $u_t = x_1 - y$.

A.2 Architecture of High-frequency Refinement Module

We detail the specific structure of the High-frequency Refinement Module as illustrated in Figure 8: Initially, the encoder feature $\mathcal{F}_{\text{enc}}$ is processed to suppress artifacts and undergo spatial transformation:

$$\mathcal{F}_{\text{trans}} = \text{ConvBlock}(\text{Denoise}(\mathcal{F}_{\text{enc}})) \quad (14)$$

To selectively integrate these details, a **Gating Mechanism** computes a pixel-wise gate map $\mathcal{G}$ to weight the transformed features. Subsequently, a **Convolutional Attention Block (CAB)** is employed to generate a spatial attention

* Equal contribution. † Corresponding author.

¹ State Key Laboratory of Multimedia Information Processing and National Engineering Research center of Visual Technology, School of Computer Science, Peking University

Algorithm 1: Training

Input: Dataset $\mathcal{D}$ with clean latents x_1 and degraded latents y .
Output: Optimized parameters θ of the conditional DiT backbone v_θ .

- 1 **repeat**
- 2 Sample $(x_1, y) \sim q(x_1, y)$ from $\mathcal{D}$;
- 3 Sample $t \sim \text{Uniform}(0, 1)$ and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$;
- 4 $x_t = tx_1 + (1 - t)y + \sqrt{t(1 - t)}\epsilon$;
- 5 $u_t = x_1 - y$;
- 6 $\mathcal{L}_{\text{fm}} = \|v_\theta(x_t, t, y) - u_t\|^2$;
- 7 $\mathcal{L}_{\text{cos}} = 1 - \cos(v_\theta(x_t, t, y), u_t)$;
- 8 $\mathcal{L}_{\text{transport}} = \mathcal{L}_{\text{fm}} + \mathcal{L}_{\text{cos}}$;
- 9 Update θ by minimizing $\mathcal{L}_{\text{transport}}$;
- 10 **until** *converged*;

map $\mathcal{A}$, which dynamically modulates the fusion of encoder and decoder features to produce the attended feature $\mathcal{F}_{\text{attn}}$:

$$\mathcal{G} = \text{Gating}([\mathcal{F}_{\text{trans}}, \mathcal{F}_{\text{dec}}]) \quad (15)$$

$$\mathcal{A} = \text{CAB}([\mathcal{F}_{\text{trans}} \odot \mathcal{G}, \mathcal{F}_{\text{dec}}]) \quad (16)$$

$$\mathcal{F}_{\text{attn}} = (\mathcal{F}_{\text{trans}} \odot \mathcal{G}) \odot \mathcal{A} + \mathcal{F}_{\text{dec}} \odot (1 + (1 - \mathcal{A}) \cdot 0.3) \quad (17)$$

where $\text{Denoise}(\cdot)$ and $\text{ConvBlock}(\cdot)$ represent the modules in Fig. 8 (b) and Fig. 8 (c), respectively. $[\cdot, \cdot]$ is the concatenation, $\odot$ is the element-wise multiplication, and $\text{Gating}(\cdot)$ denotes the gating mechanism in Fig. 8 (d). Furthermore, a **per-pixel AdaLN module** is employed to modulate the fused features. Specifically, as illustrated in Figure 8 (f), the AdaLN block takes the concatenated features as input and performs the following operations to produce the modulated feature $\mathcal{F}_{\text{mod}}$:

$$(\gamma, \beta) = \Psi([\mathcal{F}_{\text{attn}}, \mathcal{F}_{\text{dec}}]) \quad (18)$$

$$\mathcal{F}_{\text{mod}} = \text{LN}(\mathcal{F}_{\text{attn}}) \odot (1 + \gamma) + \beta \quad (19)$$

where Ψ denotes a lightweight parameter predictor consisting of two 1×1 convolutional layers and a SiLU activation function. LN represents the Layer Normalization operation. Finally, the fused features are refined through two **Res-Blocks** and integrated into the main stream via a weighted residual connection with weight w :

$$\mathcal{F}_{\text{final}} = w \cdot \text{ResBlocks}([\mathcal{F}_{\text{mod}}, \mathcal{F}_{\text{dec}}]) + \mathcal{F}_{\text{dec}} \quad (20)$$

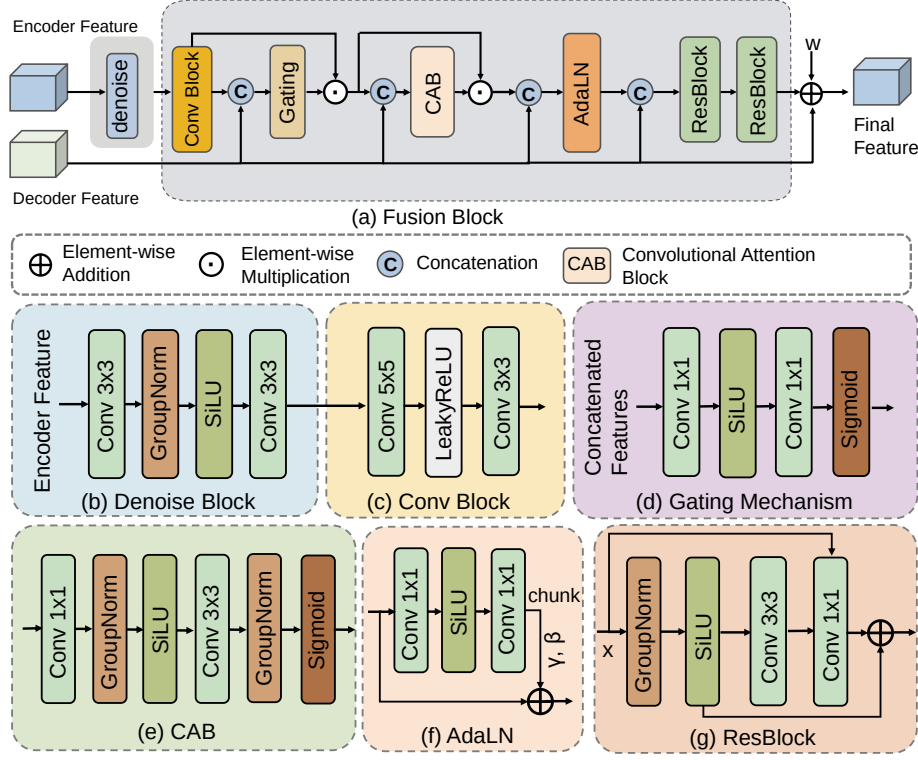


Fig. 8: The architecture of the proposed High-frequency Refinement Module. (a) Overview of the refinement pipeline. (b) Denoise Block. (c) Conv Block for spatial transformation. (d) Gating Mechanism for selective feature integration. (e) Convolutional Attention Block (CAB) for spatial attention map generation. (f) Per-pixel AdaLN for feature modulation. (g) ResBlock for final feature enhancement.

B Ablation Study

We verify each component through ablation experiments in Table 3. Starting from the baseline (first row), Degradation-Aware Latent Separation brings a significant leap, with PSNR increasing from 19.67 dB to 22.69 dB, showing that disentangling degradation in latent space provides a much cleaner prior. The Details Refinement module further improves both PSNR and SSIM. However, we observed that it might inadvertently reintroduce certain degradation artifacts. To address this, we incorporate the High-frequency Details Refinement stage, which leads to a significant drop in LPIPS to 0.114. These results demonstrate that each component plays an indispensable role in our framework.

We further compare our DWT-based loss with Focal Frequency Loss (FFL), an FFT-based frequency-domain loss. As shown in Table 4, DWT achieves comparable PSNR/SSIM while reducing average LPIPS from 0.166 to 0.114, confirm-

ing that wavelet decomposition restores local details while suppressing artifacts reintroduced by encoder feature fusion.

Table 3: Ablation study of our proposed components. The metrics are averaged over all datasets.

Degradation-Aware Latent Separation	Details Refinement	High-frequency Details Refinement	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
$\times$	$\times$	$\times$	19.67	0.691	0.336
$\checkmark$	$\times$	$\times$	22.69	0.727	0.253
$\checkmark$	$\checkmark$	$\times$	25.30	0.864	0.172
$\checkmark$	$\checkmark$	$\checkmark$	26.15	0.876	0.114

Table 4: Comparison between FFL³ and the proposed DWT loss per dataset. DWT loss achieves comparable PSNR/SSIM while substantially improving perceptual quality (LPIPS) on most datasets.

Dataset	FFL			DWT loss (Ours)		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
Rain1400 [11]	28.86	0.884	0.166	29.36	0.893	0.080
Outdoor-Rain [23]	22.92	0.824	0.173	22.50	0.815	0.187
RainDrop [42]	28.15	0.899	0.142	28.74	0.904	0.082
UHDSnow [53]	30.31	0.941	0.079	28.62	0.925	0.068
SOTS [21]	27.00	0.944	0.054	26.02	0.938	0.034
BSD68 [35]	26.71	0.835	0.311	28.36	0.858	0.178
LoL [54]	26.73	0.850	0.242	26.83	0.833	0.139
UHDM [62]	19.72	0.813	0.197	21.59	0.847	0.114
HDR-Eye [38]	25.68	0.912	0.129	23.29	0.870	0.148
Average	26.23	0.878	0.166	26.15	0.876	0.114

C Generalization and Adaptation Analysis

To further examine the generalization of SLAIR, we conduct two complementary evaluations: testing on unseen datasets without task-specific fine-tuning, and evaluating on novel degradation types beyond the original training setting.

³ FFL denotes Focal Frequency Loss: Jiang et al., “Focal Frequency Loss for Image Reconstruction and Synthesis,” ICCV, 2021.

Together they demonstrate the robustness of our degradation-agnostic latent transport.

C.1 Generalization to Unseen Datasets

We further evaluate SLAIR on three unseen degradation datasets, including RealRain⁴, Dense-Haze⁵, and EUVP⁶. As shown in Table 5, SLAIR achieves the best SSIM and LPIPS on RealRain and Dense-Haze, and the best PSNR on EUVP, indicating that its degradation-agnostic transport generalizes beyond the training datasets.

Table 5: Zero-shot results on three unseen datasets without task-specific fine-tuning.

Method	RealRain (Deraining)			Dense-Haze (Dehazing)			EUVP (Underwater)		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
AirNet [22]	19.86	0.662	0.513	10.52	0.379	0.784	19.64	0.758	0.356
PromptIR [41]	22.93	0.739	<u>0.469</u>	<u>11.40</u>	0.399	0.749	19.96	0.761	0.356
DA-CLIP [31]	23.11	<u>0.757</u>	0.484	11.04	0.393	0.763	19.51	0.764	0.351
AdaIR [8]	<u>23.03</u>	0.752	0.487	10.87	0.400	0.761	20.10	0.777	0.314
DCPT [17]	15.34	0.662	0.573	8.32	0.346	0.843	19.21	0.757	0.334
MoCE-IR [64]	20.59	0.699	0.530	11.38	0.403	0.760	20.11	<u>0.776</u>	<u>0.316</u>
VLU-Net [65]	22.09	0.742	0.496	11.46	<u>0.408</u>	<u>0.744</u>	20.12	0.769	0.321
Ours	22.82	0.795	0.450	11.09	0.435	0.685	20.43	0.772	0.332

C.2 Generalization and Adaptation to Unseen Degradations

To evaluate adaptability to degradations beyond the standard five-task training setting, we test SLAIR on the RainDrop benchmark [42] as an unseen raindrop-type degradation. We first assess the original SLAIR on both the training-task and this unseen benchmark, then fine-tune only the DiT component on raindrop data and re-evaluate to measure adaptation and retention.

As shown in Table 6, after fine-tuning only the DiT component for 6k optimization steps with a batch size of 4, SLAIR improves performance on the unseen raindrop-type benchmark while largely preserving the original five-task performance. Specifically, PSNR and SSIM remain unchanged on the five-task benchmark, with only minor increases in LPIPS and FID. This demonstrates that SLAIR can be efficiently adapted to unseen degradations with limited fine-tuning while retaining its general restoration capability.

⁴ RealRain: Li et al., “Toward Real-world Single Image Deraining: A New Benchmark and Beyond,” arXiv, 2022.

⁵ Dense-Haze: Ancuti et al., “Dense-Haze: A Benchmark for Image Dehazing with Dense-Haze and Haze-Free Images,” ICIP, 2019.

⁶ EUVP: Islam et al., “Fast Underwater Image Enhancement for Improved Visual Perception,” IEEE RA-L, 2020.

Table 6: Generalization and adaptation analysis. Performance before and after fine-tuning the DiT component on unseen-degradation data.

Model	5-task Benchmark (Avg.)				RainDrop Benchmark [42]			
	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
Original SLAIR	28.39	.882	.094	31.05	23.28	.853	.170	108.02
SLAIR w/ 6k-Step DiT fine-tuning	28.39	.882	.096	33.92	23.54	.865	.122	53.45
Performance change	+0.00%	+0.00%	+2.13%	+9.24%	+1.12%	+1.41%	-28.24%	-50.52%

Table 7: Quantitative comparison on the RainDrop benchmark. The adapted SLAIR is obtained by fine-tuning only the DiT component on the unseen-degradation data.

Method	PSNR↑	SSIM↑	LPIPS↓	FID↓
AirNet [22]	21.91	.832	.191	119.98
PromptIR [41]	22.79	.836	.174	119.66
DA-CLIP [31]	29.74	.896	.052	34.25
DiffUIR [71]	23.50	.844	.170	113.43
AdaIR [8]	23.00	.840	.174	121.01
DCPT [17]	18.28	.788	.199	118.58
MoCE-IR [64]	22.40	.829	.178	117.44
VLU-Net [65]	22.88	.837	.179	115.29
Original SLAIR	23.28	.853	.170	108.02
SLAIR w/ DiT fine-tuning	<u>23.54</u>	<u>.865</u>	<u>.122</u>	<u>53.45</u>

D HDR Data Synthesis Details

Following the pipeline illustrated in Figure 9, we construct the training set by performing data augmentation based on the collected HDR-REAL [28] and HDR-SYNTH [28] datasets:

- **Multi-sample Generation per Scene:** For each HDR scene, we first obtain a reference image through tone mapping. To simulate diverse real-world lighting conditions, we generate 10 to 14 degraded images for the same scene by applying varying exposure levels.
- **Dynamic Exposure Factor:** The intensity of the exposure is controlled by a ratio r . This ratio determines the specific percentile of the HDR image used to derive the exposure factor E , allowing for a fine-grained control over the brightness shift.
- **Degradation Simulation:** We simulate over-exposure (sensor saturation) and under-exposure (loss of shadow details) by multiplying the HDR image with the exposure factor E , followed by a clipping operation to fit the valid pixel dynamic range.
- **Validity Filtering:** Finally, we perform an automated filtering process to prune unreasonable synthesized samples. Specifically, we implement a validity check based on pixel distribution: For under-exposed images, we constrain

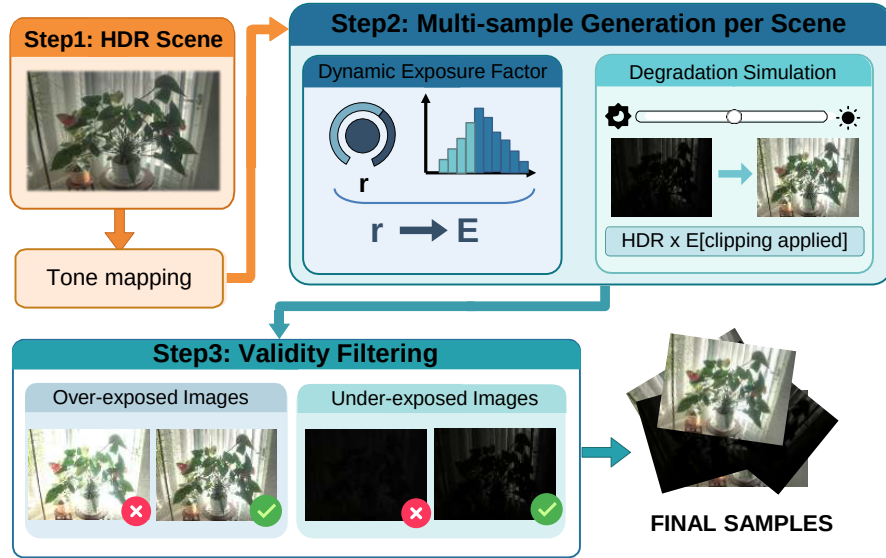


Fig. 9: The pipeline of our HDR data synthesis strategy.

the proportion of zero-valued pixels; for over-exposed images, we limit the ratio of saturated pixels (i.e., those reaching the maximum intensity). This process ensures the physical plausibility and quality of the synthesized data.

E Computational Efficiency Comparisons

Notably, as evidenced in Table 8, our proposed method allows for a substantial reduction in sampling steps with only a marginal degradation in restoration quality. While the performance peaks at 250 steps (26.15 dB in PSNR), our model already achieves highly competitive results with only 5 steps (25.53 dB). Importantly, even with this 5-step configuration, our model still outperforms all competing methods in terms of average PSNR/SSIM, indicating that the ODE-based solver efficiently captures the primary solution trajectory in the early stages of inference. The ability to trade a slight decrease in PSNR for a 50x speedup demonstrates the remarkable flexibility and practical scalability of our approach in resource-constrained environments.

As illustrated in Table 9, compared to one-pass methods, our model inevitably incurs increased overhead due to its iterative ODE-based sampling; however, it demonstrates a significant leap in efficiency relative to the SDE-based DA-CLIP [31], where our 5-step inference reduces the computational footprint (FLOPs, Runtime and FPS) by over an order of magnitude.

Table 8: Performance and efficiency trade-off across different sampling steps (T) on 512×512 resolution. PSNR and SSIM values are averaged across all evaluated datasets.

Steps (T)	FLOPs (G) ↓	Runtime (ms) ↓	PSNR ↑	SSIM ↑
5 Steps	4,853.44	685	25.53	0.861
50 Steps	35,844.24	7,240	25.73	0.863
150 Steps	104,712.70	21,852	25.73	0.863
250 Steps	173,581.16	36,225	26.15	0.876

Table 9: Computational efficiency analysis conducted on a single NVIDIA RTX PRO 6000 Blackwell GPU at a resolution of 512×512 .

Category	Model	Params (M)	Steps (T)	FLOPs (G) ↓	Runtime (ms) ↓	FPS ↑
One-pass	AirNet [22]	8.93	1	1174.77	352	2.84
	PromptIR [41]	35.59	1	690.85	133	7.53
	AdaIR [8]	28.78	1	647.02	227	4.40
	DCPT [17]	35.38	1	690.85	133	7.51
	MoCE-IR [64]	25.35	1	374.17	112	8.92
	VLU-Net [65]	188.21	1	749.39	214	4.68
Iter.	DA-CLIP [31]	295.19	100	51637.68	7755	0.13
	Ours	797.04	5 [†]	4853.44	685	1.47

[†] Sampling steps are variable. Here we report our results with $T = 5$ to highlight the superior computational efficiency of our method.

F Additional Qualitative Results

We provide additional qualitative comparisons for both Setting (i) and Setting (ii). For Setting (i), we include compact AirNet comparisons to complement the visual results, as shown in Fig. 10. For Setting (ii), we present comprehensive qualitative comparisons on five benchmark datasets, including BSD68 (Fig. 11), GoPro (Fig. 12), LoL (Fig. 13), Rain100L (Fig. 14), and SOTS (Fig. 15). These results provide a more complete visual comparison across different experimental settings.

G Resolution-Agnostic Inference and Results

High-resolution restoration is costly to train due to quadratic token complexity and memory consumption. In practical deployment, however, input resolutions vary widely, and resizing may introduce geometric distortion. We therefore design a resolution-flexible inference pipeline that (i) supports arbitrary spatial resolutions without retraining within $2\times$ training patches, and (ii) avoids resizing-induced distortion. While positional embeddings are often parameter-agnostic, an intuitive approach is to directly recompute the position embedding and RoPE based on the actual input size during inference.

Position Embedding. Given a latent partitioned into a $N = H \times W$ grid of patches (tokens), the token sequence is denoted as $\{x_n\}_{n=1}^N$, with embedding dimension d for each token.

2D Sine-Cosine Absolute Positional Embedding The position embedding $p_{h,w}$ of token $x_{h,w}$ is constructed by concatenating two 1D sine-cosine embeddings for the row and column coordinates:

$$p_{h,w} = \left[\text{PE}_{1\text{D}}(h; \frac{d}{2}), \text{PE}_{1\text{D}}(w; \frac{d}{2}) \right]. \quad (21)$$

For a scalar coordinate $u \in \mathbb{R}$, the 1D positional encoding is:

$$\text{PE}_{1\text{D}}(u; d') = \left[\sin(u\omega_0), \cos(u\omega_0), \dots, \sin(u\omega_{d'/2-1}), \cos(u\omega_{d'/2-1}) \right], \quad (22)$$

with frequencies defined as: $\omega_k = 10000^{-\frac{2k}{d'}}$, $k \in \{0, \dots, \frac{d'}{2} - 1\}$.

RoPE. The rotation matrix of RoPE is calculated as:

$$R(u) = \text{diag}(\text{Rot}(u\omega_0), \dots, \text{Rot}(u\omega_{d_h/2-1})), \quad (23)$$

with each rotation matrix $\text{Rot}(\theta)$ given by:

$$\text{Rot}(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}. \quad (24)$$

LightningDiT uses fixed 2D sine-cosine positional embeddings generated on a training grid of size $H_0 \times W_0$. At inference time, for an arbitrary token grid $H \times W$, we bicubically interpolate the positional embeddings in the 2D spatial domain and reshape them back to token sequences. This preserves the global positional prior while allowing unseen resolutions.

Dynamic RoPE Recalibration. For RoPE applied to the attention Q, K , we recompute the frequency tables based on the target (H, W) at inference. This maintains consistent relative positional modeling under resolution changes without modifying model weights.

Our framework supports inference at arbitrary resolutions without retraining. The quantitative evaluation results is shown in Table 10. The qualitative comparison also shown in Figures 16 to 23. The results shows that SLAIR is flexible for arbitrary resolutions.

Table 10: Quantitative comparison on all benchmarks under the resolution-agnostic setting. Images are proportionally scaled (long side = 512 pixels). **Bold** and underline indicate the best and second-best performance, respectively.

Method	Rain1400		Outdoor-Rain		Raindrop		UHDSnow		SOTS	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
AirNet [22]	24.84	.794	16.57	.636	22.55	.838	22.36	.772	27.54	.955
PromptIR [41]	<u>26.51</u>	.834	<u>16.91</u>	<u>.644</u>	23.05	.842	22.55	.784	29.34	<u>.975</u>
DA-CLIP [31]	27.08	<u>.840</u>	15.55	.573	30.00	.905	29.02	.920	32.03	.965
AdaIR [8]	26.31	.832	16.80	.642	23.25	.842	23.37	.819	29.05	<u>.975</u>
DCPT [17]	26.22	.834	12.13	.523	17.21	.774	20.96	.816	29.31	<u>.975</u>
MoCE-IR [64]	26.07	.820	16.73	.635	22.38	.837	23.39	.828	<u>29.43</u>	.974
VLU-Net [65]	25.92	.822	16.77	.618	22.28	.816	23.19	.830	28.65	.976
Ours	26.07	.858	20.60	.775	<u>26.27</u>	<u>.889</u>	<u>26.45</u>	<u>.878</u>	25.19	.934

Method	BSD68		LoL		UHDM		HDR-Eye		Average	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
AirNet [22]	31.26	<u>.889</u>	24.31	.789	18.35	.756	18.52	.794	22.92	.803
PromptIR [41]	31.32	<u>.889</u>	27.86	<u>.876</u>	18.48	.752	16.31	.742	23.59	.815
DA-CLIP [31]	26.45	.719	24.92	.792	16.97	.583	17.29	.795	<u>24.36</u>	.788
AdaIR [8]	<u>31.35</u>	<u>.889</u>	26.92	.869	<u>18.70</u>	<u>.760</u>	<u>18.76</u>	<u>.807</u>	23.83	<u>.826</u>
DCPT [17]	30.67	.876	28.15	.869	15.89	.577	12.42	.746	21.44	.777
MoCE-IR [64]	<u>31.35</u>	.886	28.36	.881	18.43	.755	16.51	.761	23.63	.820
VLU-Net [65]	31.41	.892	<u>28.21</u>	.868	17.20	.595	16.79	.760	23.38	.797
Ours	26.71	.831	25.12	.845	21.32	.846	25.26	.901	24.77	.862

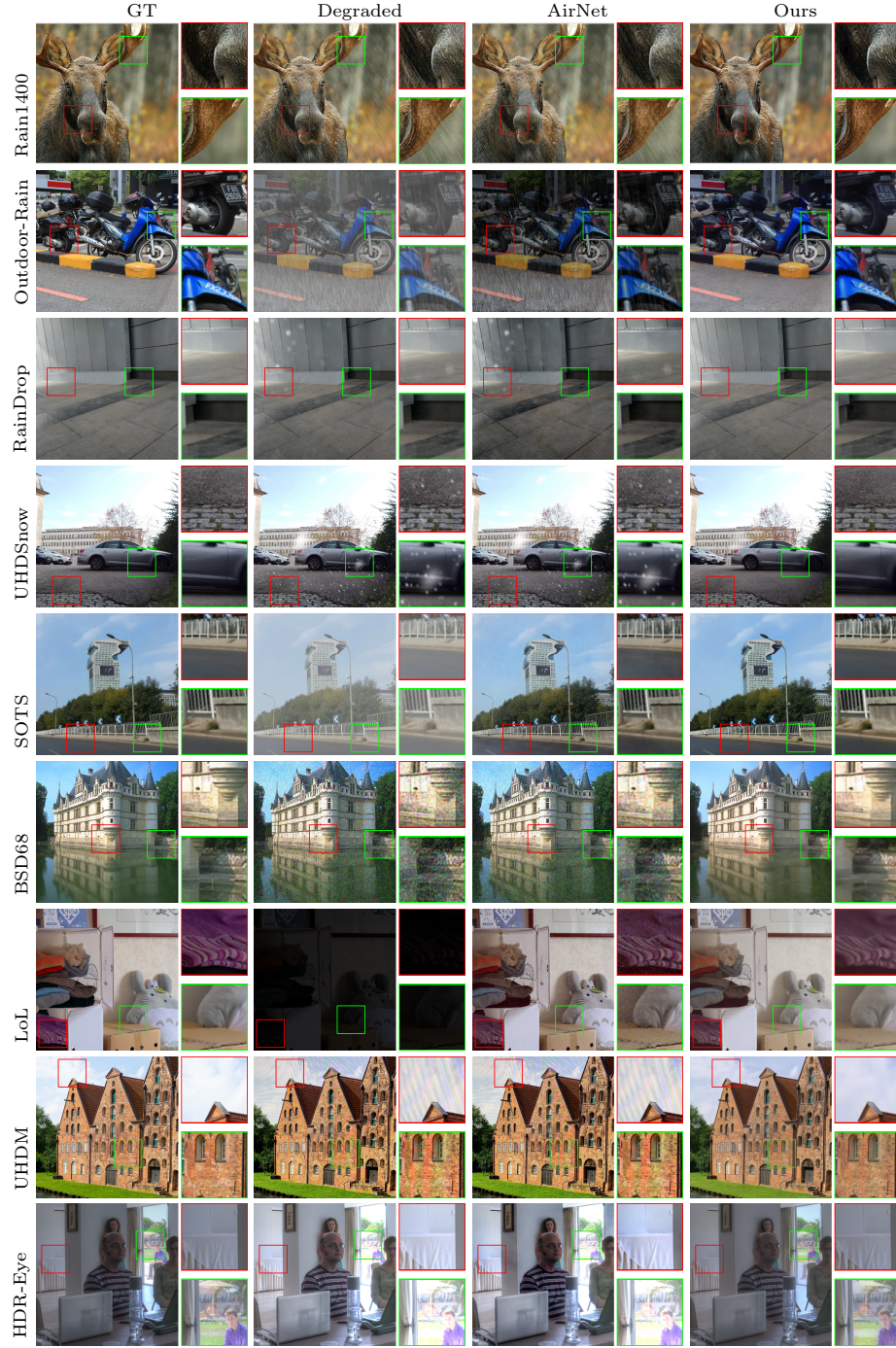


Fig. 10: Additional qualitative results of AirNet under Setting (i). Each row shows GT, degraded input, AirNet output, and our result on one dataset (Rain1400, Outdoor-Rain, RainDrop, UHDSnow, SOTS, BSD68, LoL, UHDM, HDR-Eye).

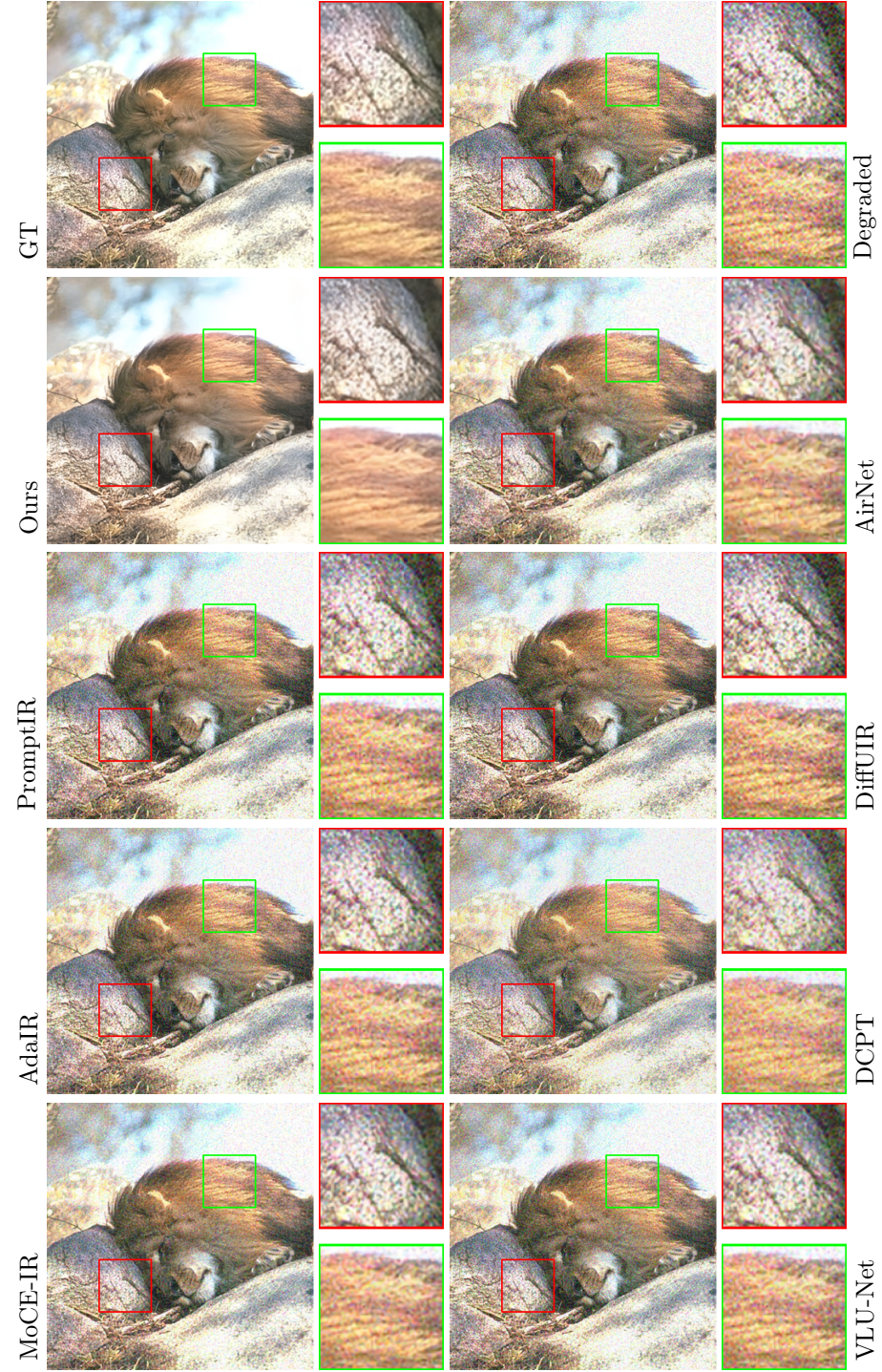


Fig. 11: Comprehensive qualitative comparison on the BSD68 dataset under Setting (ii).



Fig. 12: Comprehensive qualitative comparison on the GoPro dataset under Setting (ii).

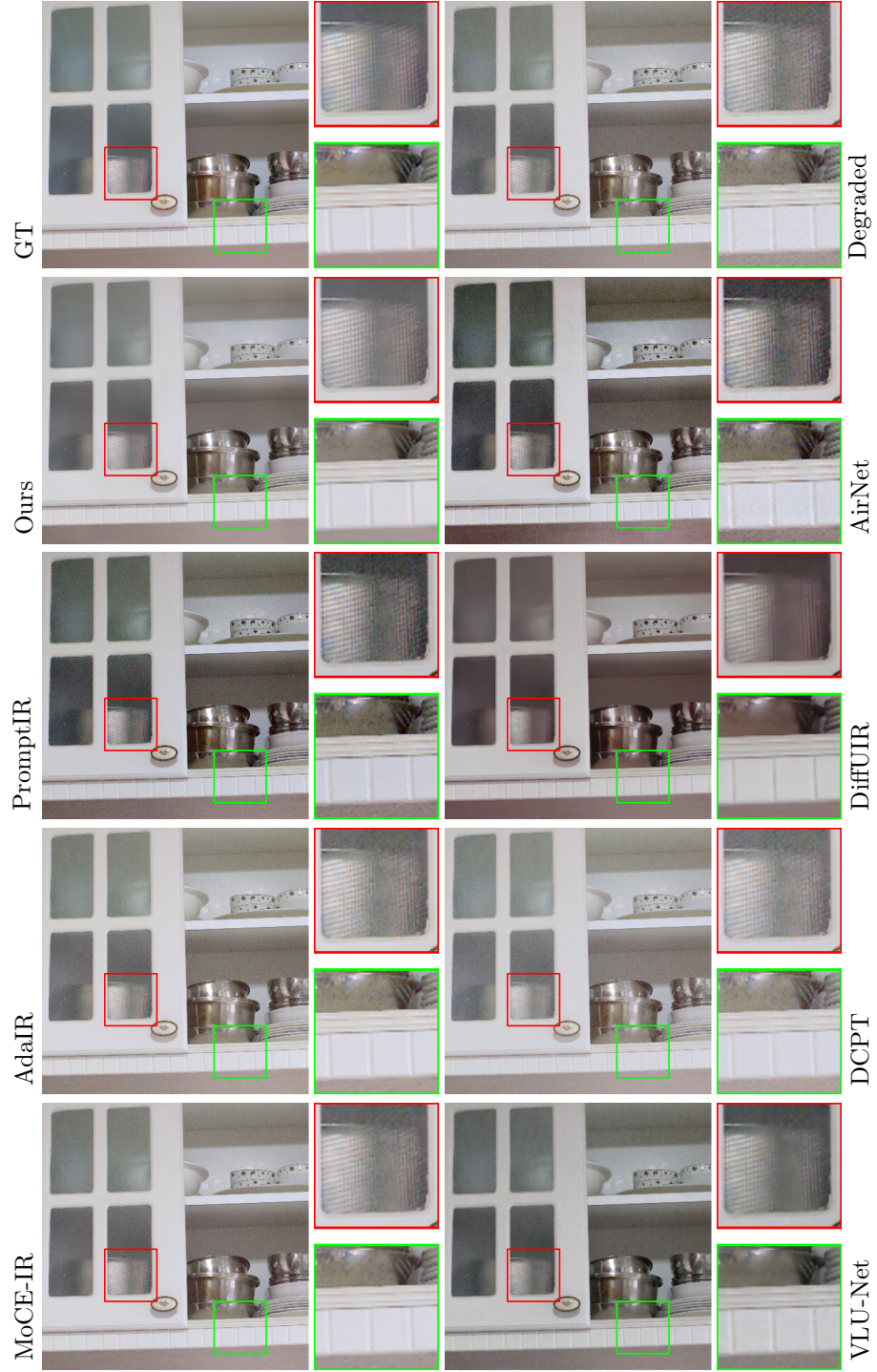


Fig. 13: Comprehensive qualitative comparison on the LoL-v1 dataset under Setting (ii).

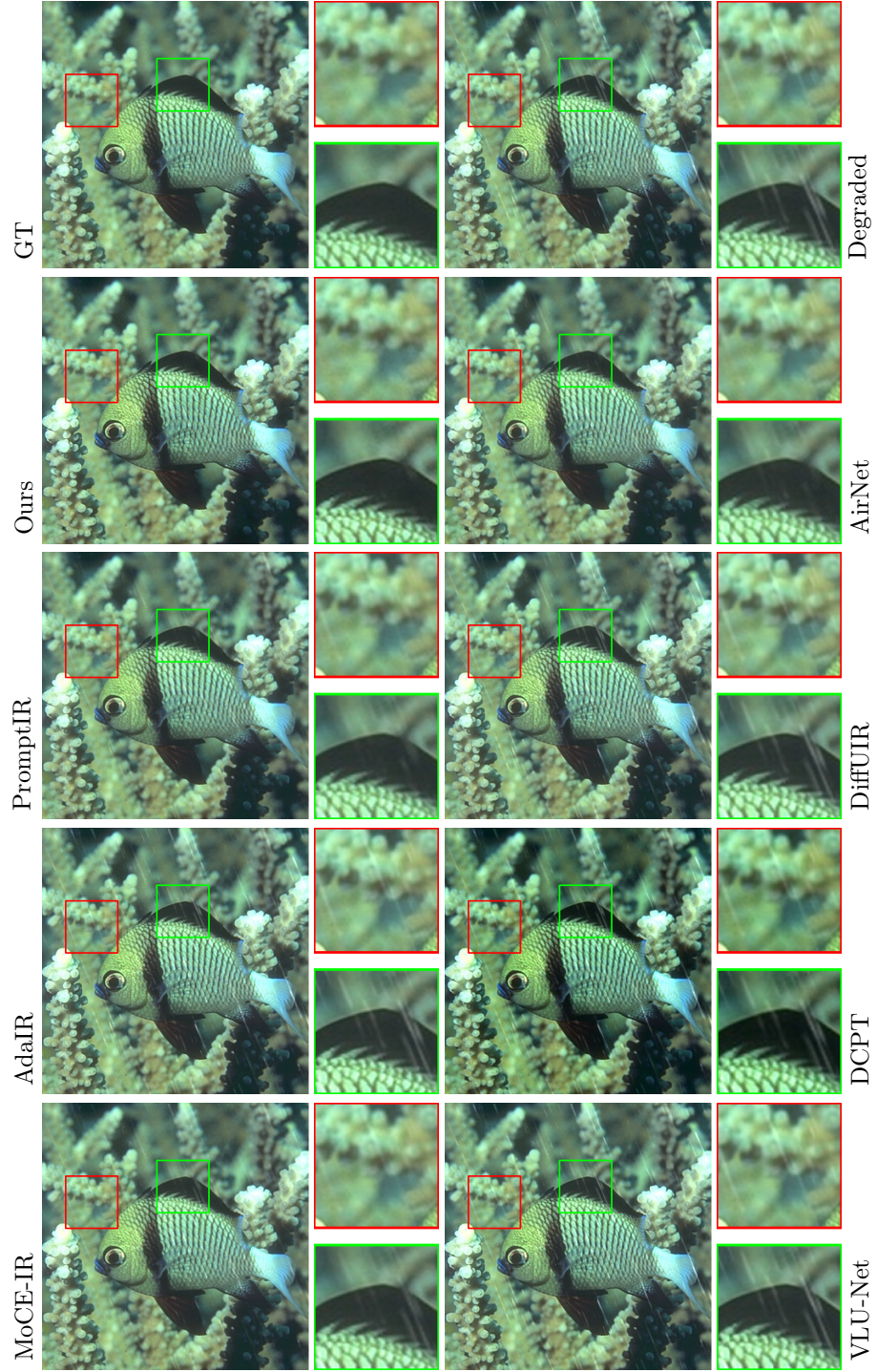


Fig. 14: Comprehensive qualitative comparison on the Rain100L dataset under Setting (ii).

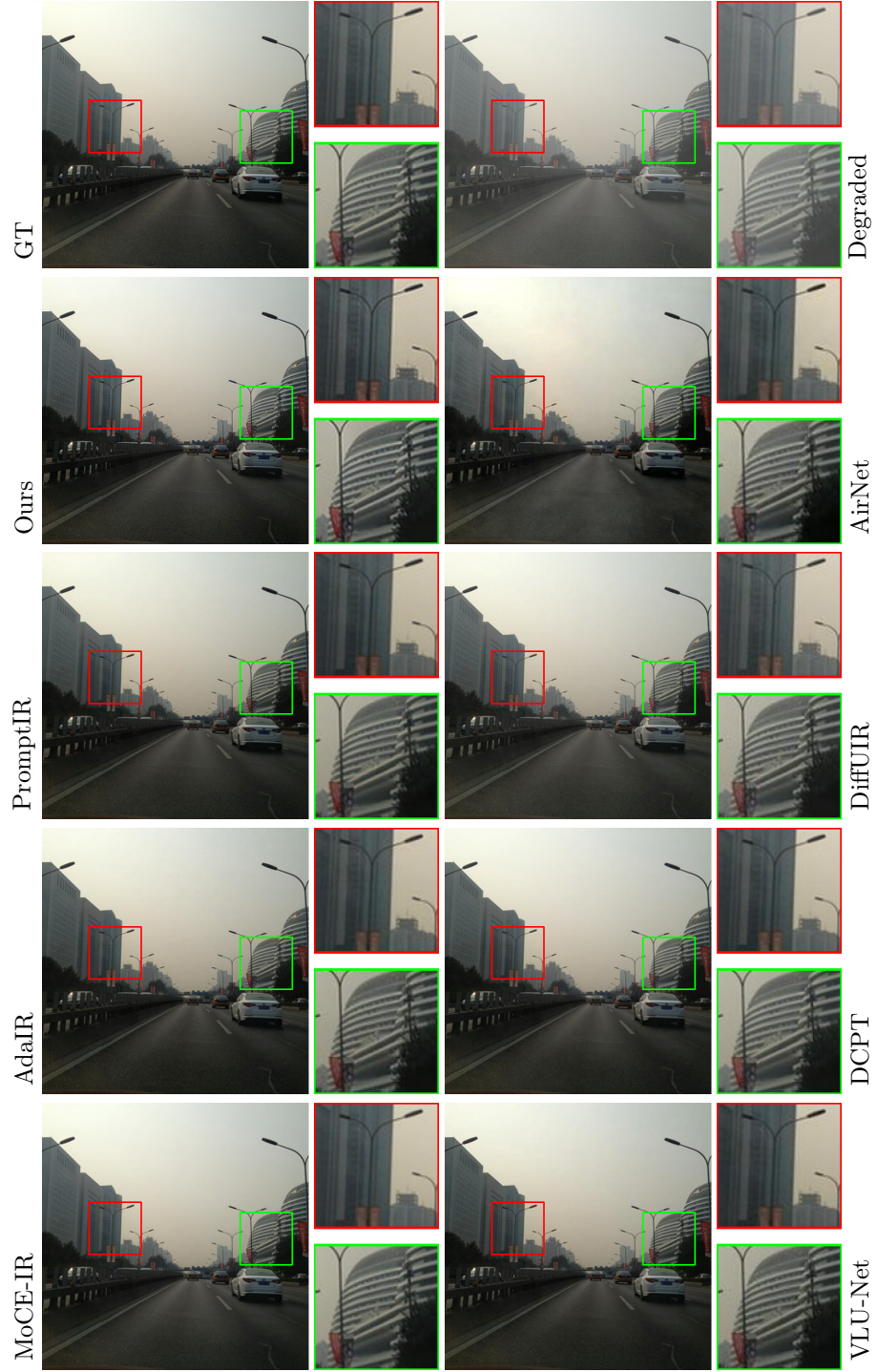


Fig. 15: Comprehensive qualitative comparison on the SOTS dataset under Setting (ii).

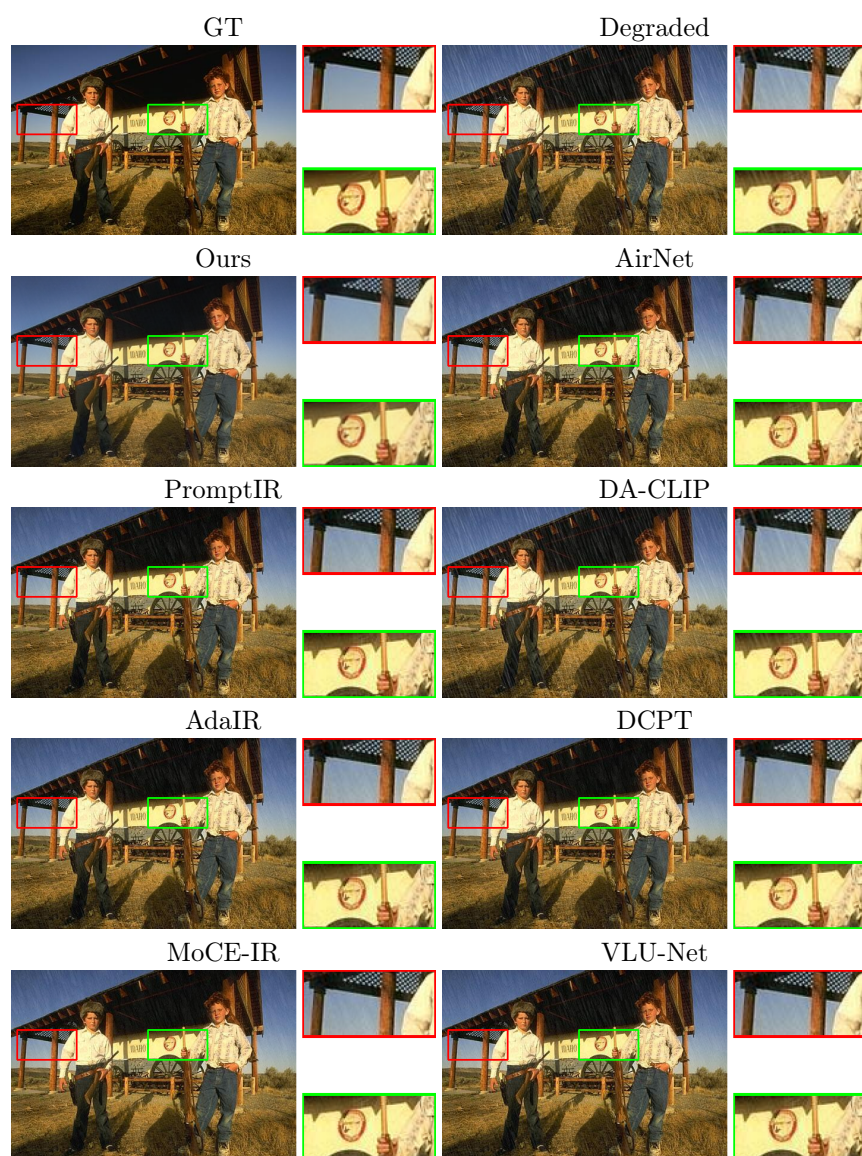


Fig. 16: Resolution-agnostic qualitative comparison on the Rain1400 dataset.

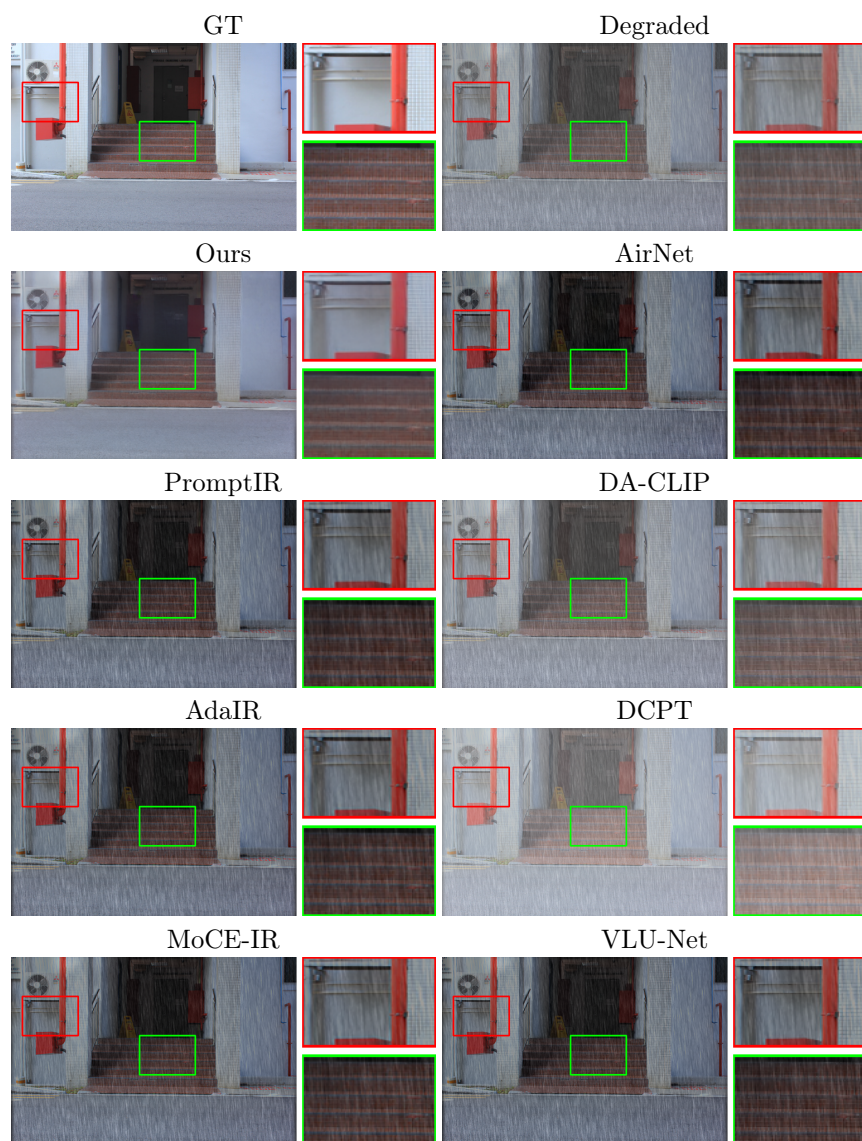


Fig. 17: Resolution-agnostic qualitative comparison on the Outdoor-Rain dataset.

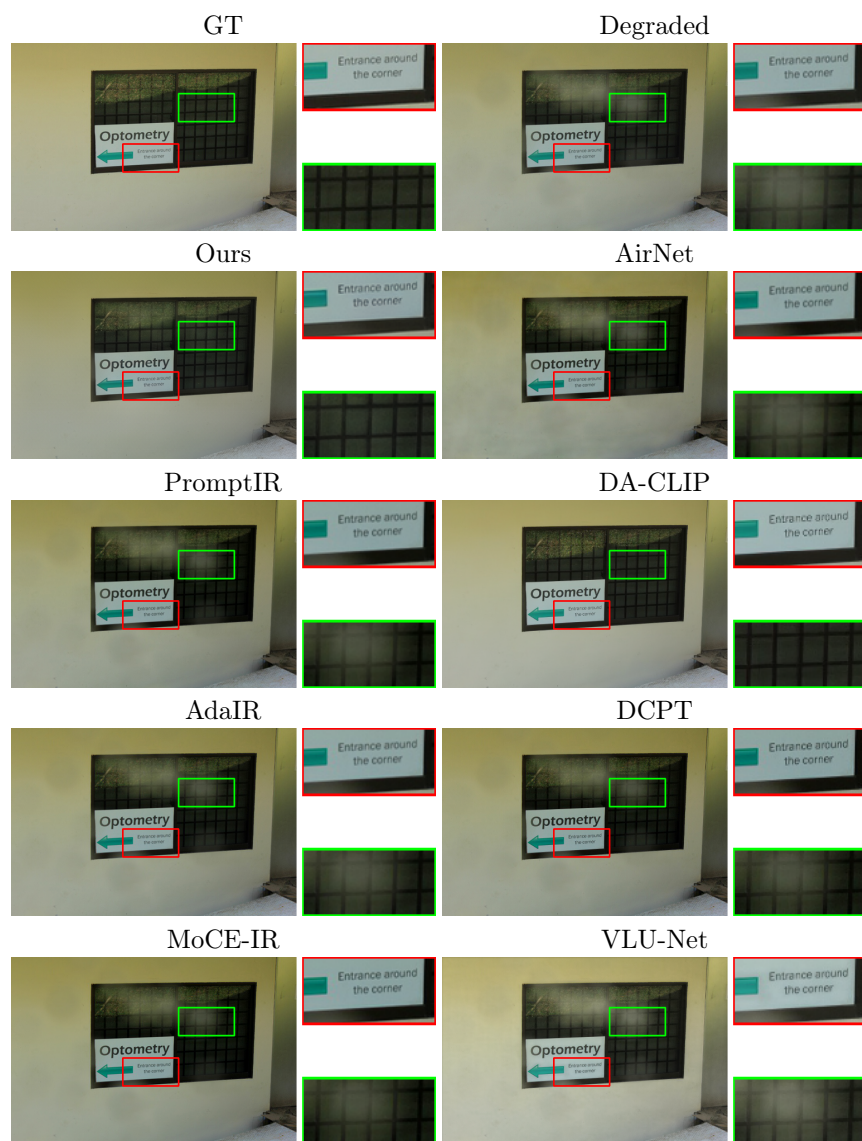


Fig. 18: Resolution-agnostic qualitative comparison on the RainDrop dataset.

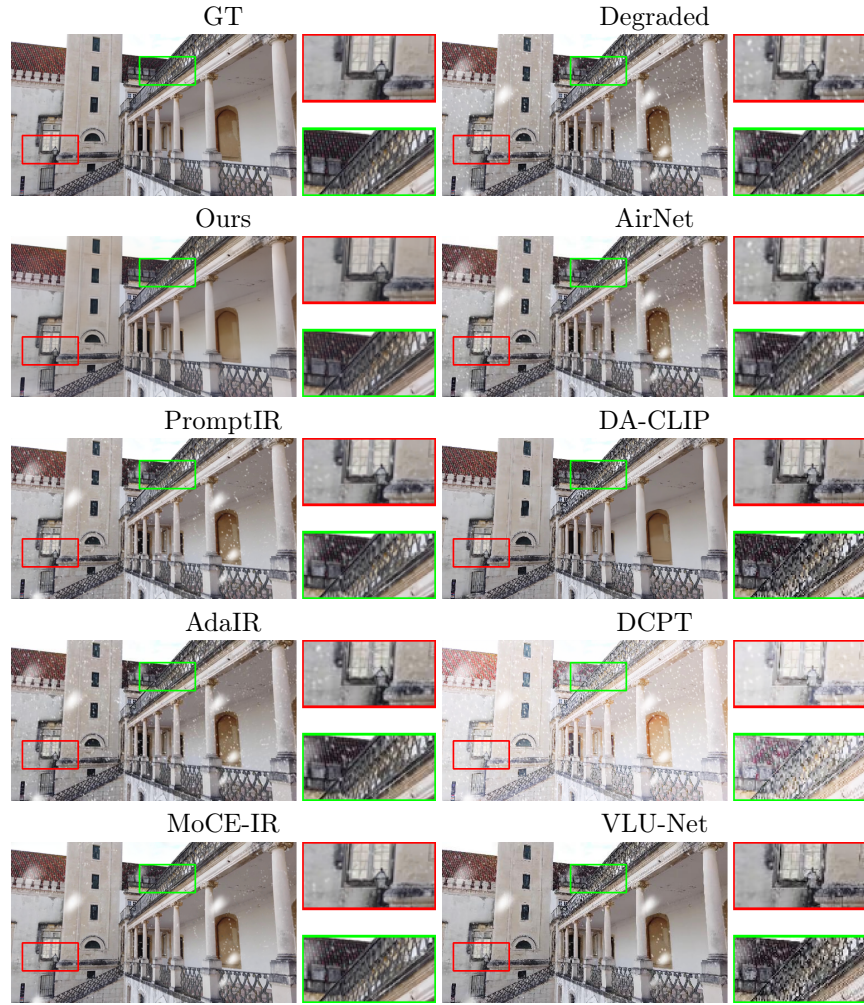


Fig. 19: Resolution-agnostic qualitative comparison on the UHDSnow dataset.

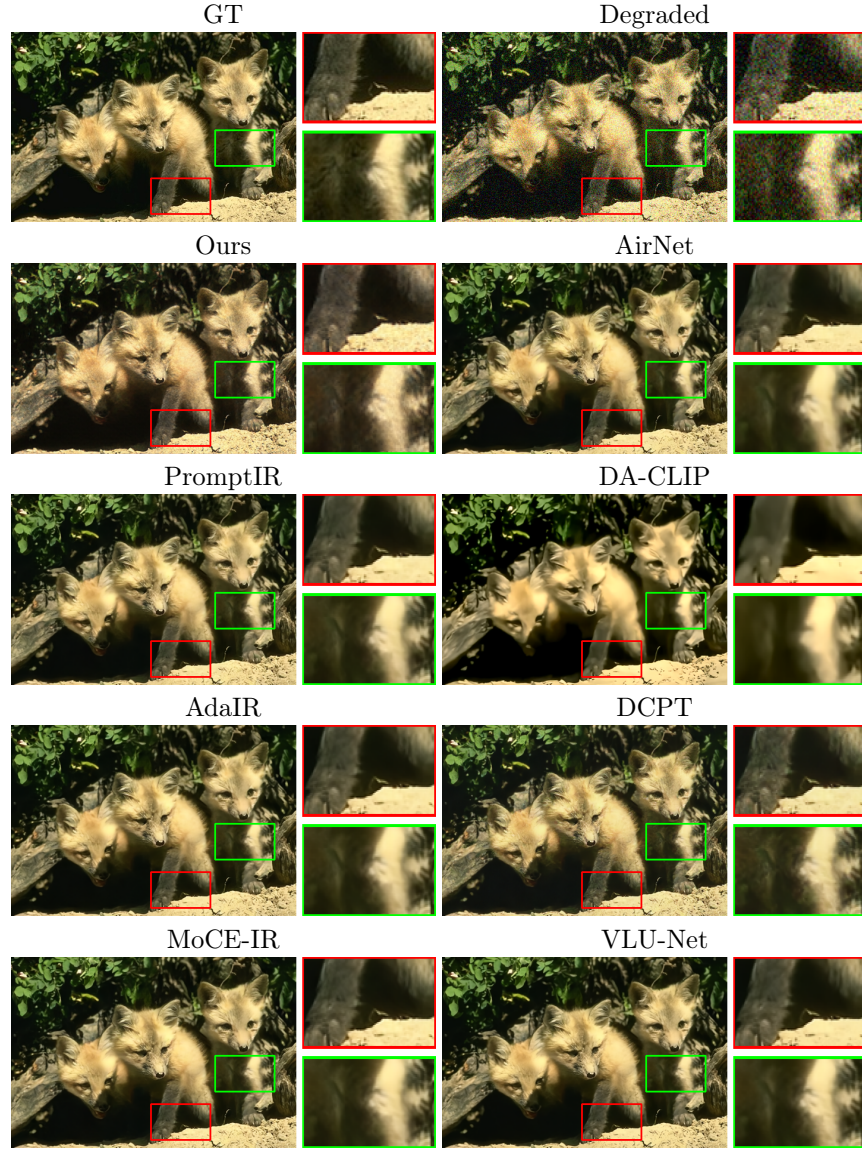


Fig. 20: Resolution-agnostic qualitative comparison on the BSD68 dataset.



Fig. 21: Resolution-agnostic qualitative comparison on the LoL-v1 dataset.



Fig. 22: Resolution-agnostic qualitative comparison on the UHDM dataset.

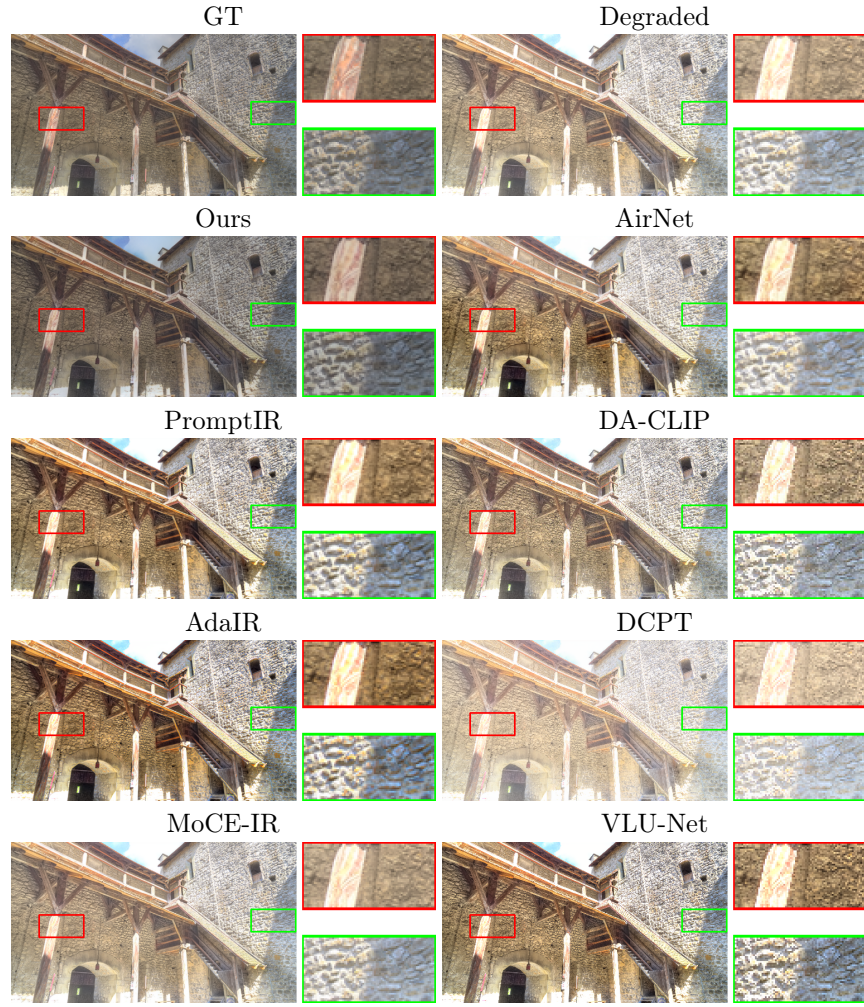


Fig. 23: Resolution-agnostic qualitative comparison on the HDR-Eye dataset.